Center for Coastline Security Technology

Year 3: Final Technical Report

Contract/PR No. N00014-05-C-0031

Prepared for
US Office of Naval Research

For the Period
3 January 2007 to 03 May 2008

Submitted by
Stewart Glegg, William Glenn, Borko Furht, P. Beaujean, G. Frisk, S. Schock, K. vonEllenrieder, P. Ananthakrishnan, R. Granata, R. Coulson

College of Engineering
Florida Atlantic University,
777 Glades Road
Boca Raton, FL 33431
561 297 3000

Submitted May 1st, 2008

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
01 MAY 2008	**N/A**	**-**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Center for Coastline Security Technology, Year 3	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Florida Atlantic University, 777 Glades Road Boca Raton, FL 33431**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES
See also ADM002086., The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**UU**	**297**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

ABSTRACT

The Center for Coastline Security Technology (CCST) focuses on research, simulation, and evaluation of coastal defense and marine domain awareness equipment, sensors and components. It builds upon the existing efforts and expertise in coastal systems and sensor research at the Institute for Ocean and Systems Engineering (IOSE), the Imaging Technology Center, and the Department of Computer Science at Florida Atlantic University.

New technologies are needed to enhance surveillance and inspections of marine activities in the coastal zone that includes major ports, small inlets, beaches, remote coastal areas and their approaches. To be efficient and cost effective it is imperative to mount the surveillance systems and sensors on autonomous platforms that can operate unsupervised for extended periods of time. The task is to effectively integrate sensors with underwater, surface and airborne autonomous and remotely operated platforms and to incorporate video and image analysis and data mining methods to quickly and effectively identify threat events.

This effort has leveraged the existing U.S. Navy marine test & evaluation facilities at the South Florida Testing Facility, which is adjacent to the major seaport at Port Everglades. This provides a unique land and aquatic test bed for the evaluation of acoustic sensors and high definition underwater and surface video mounted on unmanned fixed or mobile platforms.

This report describes the continuation of the work started in Year One and Two of the CCST project. The objective of the work in Year Three was to focus on developing technology for 3D imaging. Optical applications are based on the High Resolution Video imaging systems developed by FAU's Imaging Technology Center, and, for underwater applications, a high resolution sonar systems has been developed which can be mounted on a tetherless remotely piloted underwater vehicle.

.

In this report the details for year three of this program will be presented. The following projects are described

- The Remotely Piloted, Unmanned, Untethered, Underwater Vehicle (RPUUV)
- Acoustic Piloting, Communications and Positioning
- Environmental Assessment and Modeling: Monitoring Currents and Ambient Noise in Ports and Data Synthesis
- Development of a High Resolution Imaging Sonar for Underwater Inspections
- Experimental determination of the hydrodynamic/dynamic characteristics of a small underwater vehicle for port security
- Hydrodynamic and Dynamic Investigations for the Development of a Small Underwater Vehicle for Underwater Hull Inspection and Harbor Survey
- Chemical Sensors
- HDMAX High-Resolution QUAD HD Progressive Scan Electronic Camera System
- 3D Imaging and 3D Video Technologies for Coastline Security Applications

TABLE OF CONTENTS

LIST OF FIGURES

Figures for Section 2.2

Figures for Section 2.3

left) bit error rate (BER), (bottom right) channel impulse response, (bottom left) Doppler shift due to the relative motion between source and receiver.
Figure 2.3.19. Experimental results using QPSK modulation and a symbol bandwidth of 50 kHz: (top right) minimum mean-square error (MMSE) at the equalizer output, (top left) bit error rate (BER), (bottom right) channel impulse response, (bottom left) Doppler shift due to the relative motion between source and receiver.
Figure 2.3.20. Experimental results using QPSK modulation and a symbol bandwidth of 75 kHz: (top right) minimum mean-square error (MMSE) at the equalizer output, (top left) bit error rate (BER), (bottom right) channel impulse response, (bottom left) Doppler shift due to the relative motion between source and receiver.

Table 2.3.1. High-speed high-frequency acoustic modem message specifications.
Table 2.3.2. Measured high-speed high-frequency acoustic modem performance vs. transmission mode.
.

Figures for Section 2.4

Figure 2.4.1 All Locations Profiled Within Port Everglades
Figure 2.4.2 Spatial Variation of Sound Velocity, Temperature, and Salinity for 05-08-06 at Depth 3m
Figure 2.4.3 Spatial Variation of Sound Velocity, Temperature, and Salinity for 05-08-06 at Depth 13m
Figure 2.4.4 Current Chart For 05-08-06
Figure 2.4.5 Tide Chart For 05-08-06
Figure 2.4.6 Sound Velocity Profile for All Casts Taken On 05-08-06
Figure 2.4.7 Temperature Profile for All Casts Taken On 05-08-06
Figure 2.4.8 Specific Locations Profiled Within Port Everglades South Turning Notch
Figure 2.4.9 Sound Velocity Profiles for 05-07-07
Figure 2.4.10 Temperature Profiles for 05-07-07
Figure 2.4.11 Sound Velocity Profiles for 05-08-07
Figure 2.4.12 Temperature Profiles for 05-08-07
Figure 2.4.13 Spectrogram Showing the Power Spectral Density Over 12 Hour Sampling Period
Figure 2.4.14 Total Signal Absorption in Port Everglades South Turning Notch For 0 kHz - 1000 kHz
Figure 2.4.15 Total Signal Absorption In Port Everglades South Turning Notch For 0 kHz - 100 kHz And 100 kHz - 1000 kHz
Figure 2.4.16 Port Everglades with 4 Regions shaded in. These regions were selected by comparing turbidity profiles after all measurements had been taken.
Figure 2.4.17 Proximity to the Port inlet vs. Turbidity. Each data set is from a different day and the average tide height during the measurements is given.
Figure 2.4.18 Slope vs. Tide height. Slope is determined from the previous plot of Proximity vs. Turbidity.
Figure 2.4.19 Turbidity vs. Scattering at 535 nm.

Figures for Section 2.5

Figures for section 2.6

Figures for Section 2.7

LIST OF TABLES

EXECUTIVE SUMMARY

The Center for Coastline Security Technology (CCST) focuses on research, simulation, and evaluation of coastal defense and marine domain awareness equipment, sensors and components. It builds upon the existing efforts and expertise in coastal systems and sensor research at the Institute for Ocean and Systems Engineering (IOSE), the Imaging Technology Center, and the Department of Computer Science at Florida Atlantic University.

New technologies are needed to enhance surveillance and inspections of marine activities in the coastal zone that includes major ports, small inlets, beaches, remote coastal areas and their approaches. To be efficient and cost effective it is imperative to mount the surveillance systems and sensors on autonomous platforms that can operate unsupervised for extended periods of time. The task is to effectively integrate sensors with underwater, surface and airborne autonomous and remotely operated platforms and to incorporate video and image analysis and data mining methods to quickly and effectively identify threat events.

This effort has leveraged the existing U.S. Navy marine test & evaluation facilities at the South Florida Testing Facility, which is adjacent to the major seaport at Port Everglades. This provides a unique land and aquatic test bed for the evaluation of acoustic sensors and high definition underwater and surface video mounted on unmanned fixed or mobile platforms.

This report describes the continuation of the work started in Year One and Two of the CCST project. The objective of the work in Year Three was to focus on developing technology for 3D imaging. Optical applications are based on the High Resolution Video imaging systems developed by FAU's Imaging Technology Center, and, for underwater applications, a high resolution sonar systems has been developed which can be mounted on a tetherless remotely piloted underwater vehicle.

This document is the final report for year three of this three year program and describes the progress on the following projects

- The Development of a Remotely Piloted, Unmanned, Untethered, Underwater Vehicle (RPUUV),
- HDMAX High-Resolution QUAD HD Progressive Scan Electronic Camera System,
- 3D Imaging and 3D Video Technologies for Coastline Security Applications

The project includes the activities of nine principle investigators. The following provides a summary of the achievements of each element of the program.

Development of a Remotely Piloted Unmanned Underwater Vehicle
PI: Dr. Stewart Glegg, Project Manager: Robert Coulson
Tasks 3.1-3.5

The development of the Remotely Piloted Unmanned Underwater Vehicle is described in Section 2.2. The objective of year three of this program was to integrate and test different types of sensor systems onto the vehicles built in years one and two of this program.

The vehicle that has been developed features a vectored thruster with an 80 deg angular range, which allows the vehicle to maneuver in tight spaces. The weight of the vehicle is approximately 35 lbs and it is easily launched and recovered by a single operator from the side of a small vessel. The vehicle includes an onboard computer which processes the sensor data, the underwater video and the output from an onboard compass, pitch and roll sensor. In the vehicle developed in year one of this program, the data from these systems is relayed though a wireless RF link on the tow float to the topside console using a remote desktop capability. The vehicle is controlled through the RF link using a commercially available remote control device developed for model aircraft. In the second generation vehicle, developed in year two, control was achieved through an underwater acoustic link.

During year three the following sensor systems have been integrated and tested on the vehicles

- Obstacle avoidance sonar (section 2.2.3)
- High speed acoustic modem data link (section 2.3)
- High and low speed acoustic communications allowing for simultaneous data retrieval and vehicle control through acoustic modems (section 2.3)
- Chemical sensor (sections 2.2.4 and 2.8)
- High definition Side Looking Imaging Sonar (SLIS) (sections 2.5 and 2.2.5)

Acoustic Communications
PI: Dr. P. Beaujean
Tasks 3.6-3.8

The main objective of this portion of the project is to achieve communications for the purpose of transmitting and receiving information wirelessly between a user and the Remotely Piloted Underwater Vehicle (RPUV). Transmitted information is used to pilot the RPUV and relay its position. Information received from the RPUV combine acoustic images of the environment and status report of the vehicle. During the first year of this project radio wave (WiFi) communication was used to control the vehicle. Whenever the tow-float solution becomes impractical, a slower but fully wireless acoustic modem is to be used. The design must consider the issues associated with acoustic communications in port at high data rates, using a high-frequency acoustic modem, and the piloting and tracking of the RPUV, using a command-and-control acoustic modem.

The objectives for year 3 were the experimentation and fine tuning of the piloting (command-and-control) acoustic modem, the acoustic positioning unit and the high-frequency acoustic modem for image transmission. All the objectives have been completed:

- The piloting acoustic modem has been tested at the SeaTech marina at a maximum range of 75 m. The vehicle moved at a top speed of 0.5 m/s in 0.5 to 3 m of water. Acoustic propagation study in a port environment was also completed to aid the analysis and prediction of performance of the underwater acoustic piloting system. The model was been tested against physical measurements in the south turning notch of Port Everglades, Florida. The impulse response could be modeled with a relative echo magnitude error of

1.62 dB at worst, and a relative echo location error varying between 0% and 4% when averaged across multiple measurements and sensor locations.

- The acoustic positioning unit has been tested in a calibration tank and at the SeaTech marina at Florida Atlantic University, Dania Beach. In two meters of water and in the presence of multiple walls and boats, the estimation error in source azimuth is 0.9 degree at 20 meters.
- The high-frequency acoustic modem was installed in the remotely-piloted vehicle and transmitted snippets of canned information to the HS-HFAM receiver. These acoustic images were received at a rate of 4 per second during a set of experiments in the FAU SeaTech marina. The vehicle moved at a maximum speed of 0.5 m/s. The RPUV was piloted acoustically during this set of experiments, and no loss of performance was noticed either in terms of low-frequency acoustic piloting or in terms of high-frequency acoustic data transmission. The maximum distance from between the vehicle and the high-frequency modem receiver was 60 m.

Environmental Assessment and Modeling: Monitoring Currents and Ambient Noise in Ports and Data Synthesis

PI: Dr. George V. Frisk

Tasks 3.9-3.12

A methodology for characterizing the acoustical properties of a port environment, namely Port Everglades, has been proposed and carried out. This approach includes both a port-wide analysis of how the basic oceanographic features within the port impact the acoustic properties, and also a more focused sampling methodology within a small region of Port Everglades, allowing for the acoustic characteristics, including ambient noise, and an approximate signal absorption to be computed. The results documented through the duration of this research indicate that the temperature variation throughout the port is the principal contributor to the characteristics of the sound velocity profile. Ambient noise measurements have revealed high levels of background noise within the sub-5 kHz region, owing likely to consistent port traffic. The calculation of absorption indicates that high frequency systems, i.e. >100 kHz, may encounter problems when transmitting over a considerable distance. These are important factors for consideration when implementing a successful underwater acoustic system.

The development of an unmanned underwater vehicle at Florida Atlantic University with onboard optical sensors has prompted the temporal and spatial optical characterization of Port Everglades, with in-situ measurements of the turbidity, conductivity, and temperature. Water samples were collected for laboratory analysis where attenuation and absorption were measured with a bench top spectrometer. All of the measurements showed a high degree of variability within the port on a temporal and spatial basis. Correlations were researched between the measured properties as well as tide and current. Temporal variations showed a high correlation to tidal height but no relation was found between turbidity and current, or salinity. As a result, the planned current measurement program was not conducted. Spatial variations were primarily determined by proximity to the port inlet. Proportionality constants were discovered to relate turbidity to scattering and absorption coefficients, as well as visibility. These constants along

with future turbidity measurements will allow the optimization of any underwater camera system working within these waters.

Development of a High Resolution Imaging Sonar for Underwater Inspections
PI: Dr. S. Schock

Tasks 3.13-3.17

A high resolution focusing sidelooking sonar ("acoustic camera") was developed to generate near photographic-like images of underwater objects from maneuvering underwater vehicles. The sonar has a hemispherical acoustic projector and a 512 hydrophone line array. The resolution of the sonar is 1.5 mm in the nearfield for a center frequency of 1.6 MHz and 400 kHz of bandwidth. The resolution is substantially better (by more than a factor of 2) than commercially available sonars. Tests in the vicinity of Port Everglades, Florida demonstrated the capability of the sonar for imaging ship hulls in water with high turbidity (poor visibility). During port tests, the acoustic camera was mounted in the RPUUV which performed hull surveys and transmitted image data to the topside display computer via the RF modem. A significant result is that the acoustic camera generated high resolution imagery with a 90 degree field of view and 1.5 mm resolution while the UUV was maneuvering. Images of underwater zincs anodes mounted to ship hulls demonstrate that the sonar is capable of imaging WMD attached to ship hulls. This new acoustic imaging technology is an important development because the images produced by the acoustic camera have photographic-like resolution that allows the operator to easily recognize WMD mounted on seawalls or hulls in turbid water where visual and optical searches are not possible.

Hydrodynamic refinement and characterization of a small underwater vehicle for hull and harbor survey
PI: Karl von Ellenrieder

Tasks 3.18-3.21

A complete set of experimental measurements of the hydrodynamic coefficients on the vehicle over the entire range of operating speeds was performed in a 4' x 4' towing tank. In order to enable this work, modifications to the existing RPUUV hydrodynamic model as well as the flow facility were made. The separate contributions to the forces and moments affecting the RPUUV from the main hull and the control surfaces (propeller duct) were determined. In addition, the hydrodynamic effects of a new sonar side panel configuration for the RPUUV were examined.

Some of the key findings and future design recommendations include: 1) the propeller duct and its supporting structure contribute significantly to the overall drag of the vehicle. Modification of the strut support cross section should mitigate this effect. 2) In a turn the propeller duct produces a beneficial turning moment that counteracts the Munk moment hull. If maneuverability is found to be an issue, increasing the chord length of the propeller duct slightly could help to improve the vehicle's controllability. 3) The new sonar panel design introduces both a substantial drag penalty (C_d is roughly doubled) and a large sway force generated when

the vehicle is in a turn. If the controllability of the vehicle in turns is found to be an issue, an increase in the propeller duct length may be required.

Hydrodynamics and Dynamics Analyses of the Remotely-Piloted Unmanned Underwater Vehicle (RPUUV)
PI: Dr. P. Ananthakrishnan

Task 3.22

Year 3 efforts were focused on determining effects of

- surface waves and
- acoustic array panels (each of dimension 22 in x 7 in and to be on the sides)

on the dynamics and stability of the RPUUV. The problem formulations, modeling of wave forces and forces on acoustic panels, solution methods, simulations, new findings and contributions are presented in this report.

Froude-Krylov method is used to determine the wave force on the RPUUV. The side acoustic panels are modeled as lifting flat plate surfaces. Boundary-integral algorithm based on the Green's theorem is developed to determine the unsteady hydrodynamic coefficients. Lift and drag forces on the vehicle, appendages and fins are modeled using experimentally-determined lift and drag coefficients.

Equations governing rigid-body vehicle motion, formulated using body-fixed frame of reference, are integrated in time using the Euler's scheme to simulate vehicle dynamics. Simulations were carried out for a range of scenarios and parameter values. The vehicle is found to be quite robust and easily maneuverable with the vector thruster both in waves and with the possible addition of acoustic array panels on the sides of the vehicles.

The findings and contributions of the research to the design and development of the RPUUV are summarized in the conclusion section of the report.

Chemical Sensors
PI: Dr. Richard Granata

Task 3.23

Section 2.8 describes the formulation of a chemical method to detect underwater trace explosives, as well as the design and testing of a field-deployable device to implement the chemical method. The research goals are identified, the test materials, equipment and experiments are described and the results are discussed. The chemical compound, europium thenoyltrifluoroacetone, has been identified as an integral part of a viable underwater chemical detection method for underwater explosive traces. Included in this section is the final report on the capabilities of a chemical sensor UUV payload for detection of explosive materials for UUV applications.

High Definition Video Systems
PI: Dr. William E. Glenn

Tasks 3.24-3.27

During year 3 the Imaging Technology Center achieved all of its objectives under this program with the completion of the video compression system and solid-state recorder, integration of the final system, and the successful demonstration of a 2160-line, progressive-scan, ultra-high-definition 3D imaging system that combines a pair of FAU's HD-MAX video cameras with a pair of Sony SRX-R105 digital cinema projectors for stereo imaging and projection. Included in the system were the solid-state recorder and video compression/decompression system design. All items under design and development were completed and made ready for delivery or demonstration, as requested by the sponsor. In addition, special reports were prepared detailing different aspects of the system, including polarization optics, camera and projector setup, and JPEG-2000-based video compression processor design.

Stereo And Multi-View Image And Video Coding, Tracking, Analysis And Playback

PI: Dr. Borko Furht

Tasks 3.28-3.30

The main goal of our work was to provide semi-automatic tools to monitor marine traffic at key locations, analyzing the contents of incoming video streams, detecting potential threats, and triggering the corresponding action. Our efforts during Year 3 of the grant have been focused mostly on merging algorithms and techniques developed in the last two years of this project into a robust multi-view video surveillance system applicable to maritime domain. The developed methods for such automated surveillance are ultimately targeted for real-time or near-real-time processing from sequences obtained by both regular cameras as well as high-definition cameras supporting HDTV and/or QuadHDTV resolutions.

The objectives for Year 3 are:

- *Develop effective methods to 3D video compression, delivery and playback.* Techniques and methods for efficient compression of stereo and multi-view sequences is an ongoing research area. It is anticipated that 3D video improves surveillance applications. 3D autostereoscopic displays (no glasses required) are recently being released and are becoming notably inexpensive. The goal is also to create a 3D video player for Sharp autostereoscopic display, which is one of the first commercially available autostereoscopic displays.
- *Develop purely computational as well as biologically plausible methods and algorithms for detection, tracking and classification of video objects using depth information acquired from multiple cameras.* In addition to investigating classical, purely

computational models for detection of video objects, we also studied models inspired by principles of human visual attention. While a single-view object segmentation is limited in the sense that the occlusion is difficult to detect and it is difficult to distinguish far objects from close ones, segmentation with depth information allows for easy occlusion detection, helps distinguishing far from close objects, and helps the task of classification and tracking. Depth information provides an important feature of detected video objects that can be used for further analysis.

The following are short summaries of the projects related to these objectives.

A Practical Rule-Based Multiple Object Tracking System for Traffic Surveillance

We have developed a novel and effective rule-based multiple object tracking system for traffic surveillance using a collaborative background extraction algorithm, which collaboratively extracts a background from multiple independent extractions to remove spurious background pixels. The multiple object tracking is based on differenced binary images between video frames and the extracted backgrounds and is therefore simplified. The rule-based strategies are applied for thresholding, outlier removal, object consolidation, neighboring objects separation, and shadow removal. Empirical results show that our multiple object tracking system is highly accurate for traffic surveillance under conditions of occlusion and background variations.

A Progressive Edge-Based Stereo Correspondence Method

Local stereo correspondence is usually not satisfactory because neither big window nor small window based methods can accurately match densely-textured and textureless regions at the same time. In this paper, we present a progressive edge-based stereo matching algorithm, in which big window and small window based matches are progressively integrated based on the edges of a disparity map of a big window based matching. In addition, an arbitrarily-shaped window based matching is used for the regions where big windows and small windows can not find matches, and a novel optimization method, progressive outlier remover, is used to effectively remove outliers and noise. Empirical results show that our method is comparable to some state-of-the-art stereo correspondence algorithms.

Multiview Video Compression and Navigation

*Usi*ng multiple cameras or arrays of cameras for surveillance applications improves monitoring effectiveness. Use of multiple cameras, however, makes navigation complex and increases the amount of data to be processed. We have developed multiview data capture and compression using multiview encoder. The multiple views are navigated using a motion sensing remote control. The video data is compressed using multiview video encoder that exploits dependencies among the multiple views to improve compression.

Smart Video Encoding

Encoding video frames blindly leads to unnecessary computation and wasted bandwidth. A typical video frame has a few areas of interest, with most of the video data representing

background or regions of lower importance. Smart video encoding identifies areas of interest and encodes them with higher quality thereby focusing resources on few areas of interest. We have developed an approach to perceptual quality enhancement based on content awareness relative to points of salient locations in video pictures. We detect the visually salient regions of interest and modify the rate control through quantization based not only on buffer fullness but also on the level of detail in the picture. The results show that by making small quality compromises in regions of the picture that are perceptually less important and in such a way that is intended to be minimally perceptible, we can improve the perceived quality of the selected regions in the video without affecting the bitrate. The experimental results show that the proposed method improves perceptual quality of salient regions and decreases the bitrate with some loss in overall PSNR.

3D Video Compression

We have developed algorithms for compression 3D video for new generation of autostereoscopic displays using asymmetric view coding. Asymmetric view coding encodes the stereo views with different quality. It has been shown that the human visual system is able to compensate for this asymmetric view quality and present a good quality 3D video. Asymmetric video coding can be exploited to reduce the bandwidth requirements for 3DTV services. The key factors that affect the asymmetric video coding are the compression algorithms, the human visual system, and the 3D display. We conducted a subjective evaluation of 3D video with asymmetric view quality and encoded using MPEG-2. We also studied the impact of eye dominance on the perceived quality. We show that asymmetric view coding can be used to reduce the bandwidth requirements of 3DTV services based on MPEG-2 view coding.

Robust Detection and Tracking of Video Objects in Stereo for Smart Video Surveillance

This section presents a design of a system for video surveillance employing object detection and tracking which integrates depth information from a pair of cameras. It is a part of a smart maritime video surveillance system in which robustness and near real-time processing are among the major design goals. A robust surveillance system must aim to produce a minimal amount of false positive results while simultaneously keeping the number of false negatives as low as possible. Furthermore, such a system must be able to track both rigid and non-rigid objects in complex environments and overcome automated tracking difficulties that arise due to object occlusion. Unlike many other surveillance systems, our system uses stereo video footage to estimate depth information, improving the quality of object detection and tracking. The method consists of object segmentation based on a novel class of Bayesian probabilistic neural networks, computation of a depth map, and object tracking based on feature descriptors including intensity, color, shape, motion and depth. Experimental results are provided to demonstrate the performance of our approach.

Design and Implementation of an Optical Flow-based Autonomous Video Surveillance System

This section presents the design of a surveillance system based on optical flow. Considerations of the capabilities, limitations, and possible solutions to these limitations are presented. Additionally, an evaluation of the performance of optical flow in situations such as depth estimation, rigid classification, non-rigid classification, segmentation, and tracking will be

presented. Our main contribution is a new system level architecture based on one algorithm for an entire video processing system. The case study is a video surveillance system, whereas optical flow is the main core.

Using a Computational Model of Human Visual Attention for Detecting Objects in Images

Computational models of human visual attention describe the early processes of human vision by predicting the areas of an image that are likely points of fixation. In this work we analyze the suitability of a computational model of human bottom-up visual attention for detecting salient regions of interest in images. The performance of using the predicted salient points to detect regions of interest is evaluated. Our results suggest that the points of attention generated by such models provide a principled, unsupervised, biologically-inspired method for extracting seeds which can be subsequently used by region growing segmentation algorithms.

Investigation of the Suitability of Using a Computational Model of Visual Attention for Detecting Objects of Interest in Video Surveillance Footage

In section work we investigate the suitability of using points of attention generated by a computational model for detecting objects of interest in video surveillance footage. The computational model was used to generate points of attention for thirty video surveillance sequences. The results were then analyzed and discussed, with specific recommendations made as to how to improve performance. We recommend empirically determining a threshold of the voltage (intensity) of points to discard and creating a post-processing block for discarding points that target areas of the frame that do not exhibit movement.

1.0 INTRODUCTION

1.1 Overview

1.1.1 Background

The Center for Coastline Security Technology (CCST) focuses on research, simulation, and evaluation of coastal defense and marine domain awareness equipment, sensors and components. It builds upon the existing efforts and expertise in coastal systems and sensor research at the Institute for Ocean and Systems Engineering (IOSE), the Imaging Technology Center, and the Department of Computer Science at Florida Atlantic University.

New technologies are needed to enhance surveillance and inspections of marine activities in the coastal zone that includes major ports, small inlets, beaches, remote coastal areas and their approaches. To be efficient and cost effective it is imperative to mount the surveillance systems and sensors on autonomous platforms that can operate unsupervised for extended periods of time. The task is to effectively integrate sensors with underwater, surface and airborne autonomous and remotely operated platforms and to incorporate video and image analysis and data mining methods to quickly and effectively identify threat events.

This effort has leveraged the existing U.S. Navy marine test & evaluation facilities at the South Florida Testing Facility, which is adjacent to the major seaport at Port Everglades. This provides a unique land and aquatic test bed for the evaluation of acoustic sensors and high definition underwater and surface video mounted on unmanned fixed or mobile platforms.

This report describes the continuation of the work started in Year One and Two of the CCST project. The objective of the work in Year Three was to focus on developing technology for 3D imaging. Optical applications are based on the High Resolution Video imaging systems developed by FAU's Imaging Technology Center, and, for underwater applications, a high resolution sonar systems has been developed which can be mounted on a tetherless remotely piloted underwater vehicle.

1.1.2 Technical Objectives

As time progresses it is becoming increasingly apparent that providing elevated homeland security in ports and harbors is limited by operational costs. Budgets for port security are several times larger than they were before the events of 9/11/01 and cost is now a major issue for both federal and local agencies. Furthermore, when Navy ships dock in areas also used for civilian activities, security issues are more complex and require close collaboration between all agencies involved. The same principles apply in overseas ports, as evidenced by the attack on the USS Cole, and port security technology, which is portable to international locations, has an important role in force protection.

Given these prerequisites it is the primary objective of this program to develop new technology for port security that provides unique capabilities for security inspections, threat detection and

rapid response, at lower operational costs. To achieve this, attention has been focused on technologies in which the members of the center have existing expertise, with the intent of turning these technologies into operational systems in a three year program.

The technologies that have been developed in this program are:

1) *Underwater vehicles for survey and inspection*: In the CCST program a low cost, one man operated, remotely piloted unmanned, untethered, underwater vehicle, has been developed which will provide real time underwater video and sonar images to a topside console. The specific application to be addressed is underwater inspections by rapid response teams, and routine inspection activities, currently carried out by scuba divers. This technology is intended to reduce the need for divers on a 24/7 basis. During year one of the program a vehicle was developed with a tow float and a RF antenna to provide the underwater video and sonar data to a topside console. In Year Two a tetherless capability was added by replacing the tow float RF antenna with a high speed acoustic modem. The objective of year three of this program has been to develop and test the operation of a new sonar, in a 2D imaging configuration, mounted on the vehicle in such a way that its performance is optimized. The optimum configuration of the sonar mounting system, including the hydrodynamic design of the vehicle, has been determined during the program of work for year three.

   A key to the success of the vehicle is the use of a high speed acoustic communication system to link the topside operator to the vehicle. The system needs to transmit control commands to the vehicle at the same time as sonar and video images are being transmitted to the topside. In Year Three of this program the acoustic communication system was optimized and tested for this specific application.

   In addition, the chemical sensors developed in years two and three of the program have been mounted to the vehicle and tested to determine if real time chemical analysis can be achieved.

   The successful operation of the underwater vehicle will be determined by local conditions which depend on acoustical, oceanographic, and optical properties of the water column. These will vary significantly, both within a particular port and at ports in different locations. In Years One and Two of the project a study was started to characterize the environment in which the vehicle will operate, and this has been completed in Year Three.

2) *High Definition Video Systems*: High definition video cameras provide an order of magnitude improvement in field of view and/or range over and above conventional systems. Consequently they are a necessity for harbor surveillance, but their implementation in this environment is limited by size and cost. At Florida Atlantic University's Imaging Technology Center, a compact high definition camera has been developed and is ready for the commercial market, the primary customers being the film industry. For the port security application there are several research issues that still need to be addressed, specifically, managing the high data output rate of the camera, and

testing the camera in the marine environment. The test and evaluation issue has been addressed by the ITC in collaboration with NAVSEA Carderock's South Florida Test Facility, which has towers overlooking Port Everglades, and the adjacent inlet, which are already used by the USCG for video surveillance.

During Year One of the program, the ITC completed the HDMAX camera system 3840x2160 format progressively scanned at continuously variable frame rates up to 30 FPS at full resolution. This format has four times as many resolvable pixels as the HDTV system being broadcast at present. The reason for choosing this format for NAVY purposes is that, with minor interface modifications, it is compatible with the infrastructure that is presently available. Displays, recorders and transmission links are available to handle the output of these cameras. Recording, transmission and display has been demonstrated in previous year's programs.

In Year Two of the program the display adapter was modified so that it could drive the Sony 4K theatre projectors. Two such systems were fabricated and two projectors ordered. Also fabricated were two solid-state recorders that had been designed in Year One.

The 3-D theatre display system built in Year Two required a special screen in order to view the images with Polaroid glasses. Normal screen material destroys the polarization. The ITC has tested the performance with a smaller screen that does not destroy the polarization; the lab purchased a full size screen on the Year Three project.

During Year Two it was found that the black level of the camera drifted with temperature. Consequently in Year Three, a temperature controller was developed to provide stable operation. The solid-state recorder designed in Year One was monochrome. In Year Three, this design was upgraded to color.

3) *Imaging and Video Technologies:* In this part of the project, carried out in the Department of Computer Science and Engineering, the project has continued to investigate and develop techniques, technologies, and algorithms needed to create and analyze 3D images and 3D videos provided by multiple HDTV cameras with the specific focus on coastline security applications. The following three projects were performed.

   a. Visual Surveillance of Object Motion and Behaviors: This project focuses on the extension of prior work towards a complete visual surveillance solution, with emphasis on detecting suspicious behavior, and on the use of biologically-inspired models of human visual attention to detect objects of interest within a scene, magnify them for viewing purposes, and guide the human operator's attention to them.
   b. Coastline Surveillance and Event Analysis: This addresses the task of surveillance and event analysis for coastline scenes. Cameras located at strategic positions are used to continuously collect video data at coastlines. The video data are automatically analyzed to understand the patterns of events, and most importantly, raise alarms for abnormal events.

c. Efficient Video Compression and Communication for Surveillance Applications: With the decreasing cost of cameras, it is possible to use multiple cameras or arrays of cameras to capture immersive video. The 3D video applications can be seen as a sub-set of multi view video with two selected views for 3D rendering. As the number of cameras increases, the amount of storage and bandwidth required will increase tremendously. Applications such as surveillance require storage of video over extended periods of time and efficient compression algorithms are necessary. The multiple views of a scene usually have substantial similarity that can be exploited to increase the compression ratio for multi-view video coding. Effective use of prediction is the key to compressing multi-view video. We have developed a hyper-cube based prediction for multi-view video that can improve prediction efficiency with minimal overhead.

In the following sections the details of each part of this program will be described, including the following projects:

- The Remotely Piloted, Unmanned, Untethered, Underwater Vehicle (RPUUV),
  PI: Dr. S. Glegg
- Acoustic Piloting, Communications and Positioning
  PI: Dr. P.Beaujean
- Environmental Assessment and Modeling: Monitoring Currents and Ambient Noise in Ports and Data Synthesis
  PI: Dr. George V. Frisk
- Development of a High Resolution Imaging Sonar for Underwater Inspections
  PI: Dr. Steven Schock
- Hydrodynamic refinement and characterization of a small underwater vehicle for hull and harbor survey
  PI: P. Ananthakrishnan & Dr. von Ellenrieder
- Chemical Sensors
  PI: Dr. Richard Granata
- HDMAX High-Resolution QUAD HD Progressive Scan Electronic Camera System,
  PI: Dr. W. Glenn,
- 3D Imaging and 3D Video Technologies for Coastline Security Applications
  PI: Dr. B. Furht

2.0 The Remotely Piloted, Unmanned, Untethered, Underwater Vehicle (RPUUV)

2.1 Summary

Currently unmanned underwater vehicles fall into two distinct classes: (1) Remotely operated vehicles that are tethered to a topside operations console. These devices are usually cage like and are designed to have a hovering capability for close up inspection of a site. They are limited by the necessity to drag a tether with them and so are rarely used for large area rapid surveys. (2) Autonomous Underwater Vehicles that are untethered and used extensively to carry out large area surveys for MCM applications. These vehicles are given a pre specified set of tasks, and have an on board navigation and object recognition capability. They are limited because currently they do not provide real time video or sonar images to the topside operator and rely on either on board intelligence or post deployment analysis to detect and evaluate targets. The autonomous capability requires sophisticated on board sensors that drive up the cost.

An alternative approach combines the remotely piloted features of an ROV with the advantages of the untethered AUV. To achieve this requires a high speed wireless underwater communications capability that is only just beginning to become available. It was proposed to develop this type of vehicle and the enabling technology as part of this program and the details of the vehicle development and test program will be described in section 2.0.

At the present time underwater wireless communication is carried out acoustically by use of an acoustic modem. Florida Atlantic University has had a strong program on acoustic modem research for the past ten years, and has developed a number of acoustic communication devices that are used on fully operational AUVs. However to achieve remotely piloted vehicle operation it was not clear at the beginning of this project that acoustic devices will be able to provide the communication rates needed, especially in a shallow water harbor environment. Progress on the development of this technology is described in section 2.3, including a description of how the technology has been used to transmit video images through an acoustic link to a topside console.

For an RPUUV to achieve its full potential an on board suite of sensors will be required which include high definition side looking sonars, integrated obstacle avoidance sonars, and chemical sensors. In the third year of this program we have continued studies on each of these sensor packages, and have integrated them into an RPUUV. These systems are described in section 2.2, 2.5 and 2.8. In addition good hydrodynamics with the sensor systems attached are required for stability and lower power consumption and research in these areas are described in sections 2.6 and 2.7.

The operation of the communication system and other sensors will also depend on the details of the port environment, including speed of sound profiles, turbidity profiles and the local currents. A study to investigate these features in Port Everglades is described in section 2.4.

2.2 Development of a Remotely Piloted Unmanned Underwater Vehicle

PI: Dr. Stewart Glegg, Project Manager: Robert Coulson

Tasks 3.1-3.5

2.2.1 Summary

The development of the Remotely Piloted Unmanned Underwater Vehicle is described in Section 2.2. The objective of year three of this program was to integrate and test different types of sensor systems onto the vehicles built in years one and two of this program.

The vehicle that has been developed features a vectored thruster with an 80 deg angular range, which allows the vehicle to maneuver in tight spaces. The weight of the vehicle is approximately 35 lbs and it is easily launched and recovered by a single operator from the side of a small vessel. The vehicle includes an onboard computer which processes the sensor data, the underwater video and the output from an onboard compass, pitch and roll sensor. In the vehicle developed in year one of this program, the data from these systems is relayed though a wireless RF link on the tow float to the topside console using a remote desktop capability. The vehicle is controlled through the RF link using a commercially available remote control device developed for model aircraft. In the second generation vehicle, developed in year two, control was achieved through an underwater acoustic link.

During year three the following sensor systems have been integrated and tested on the vehicles

- Obstacle avoidance sonar (section 2.2.3)
- High speed acoustic modem data link (section 2.3)
- High and low speed acoustic communications allowing for simultaneous data retrieval and vehicle control through acoustic modems (section 2.3)
- Chemical sensor (sections 2.2.4 and 2.8)
- High definition Side Looking Imaging Sonar (SLIS) (sections 2.5 and 2.2.5)

2.2.2 Introduction

The main objective of this task is to develop the platform that will support the sensors being developed in the other parts of the project. During the first year of the program a vehicle was developed which has enabled further developments of sensors and communication systems. The first generation vehicle was attached to a tow float with an RF antenna through which the vehicle communicates with a topside console. This system has provided valuable information on the operational requirements for vehicle deployment, and, during year two and three has been modified significantly and tested in both open ocean and port environments. The major task for year two of the program was to build a second generation vehicle, and to develop the designs for the attachment of the high resolution sonar, and the chemical sensor. The second generation vehicle differs

from the first generation because it is controlled though an acoustic link which introduces a number of additional challenges. During year three of this program an number of sensor systems have been integrated and tested in the vehicle. These include the obstacle avoidance sonar, the high speed acoustic modem, the chemical sensor and the Side Looking Imaging Sonar (SLIS).

2.2.3 Integration and Testing of the Obstacle Avoidance Sonar (Tasks 3.1 and 3.2)

2.2.3.1 Sonar Integration

The PC-View scanning sonar which was installed on the vehicle in Year One of the program was replaced with an 8 channel obstacle avoidance sonar package during year three. A CAD model of the obstacle avoidance package with a nose-cone accommodating the 8 transducers is shown in Figure 2.2.1. Packaging of the obstacle avoidance sonar electronics are also shown in this figure.

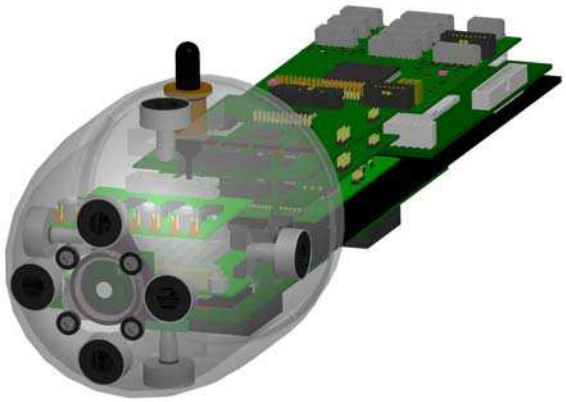


Figure 2.2.1: Obstacle Avoidance Sonar Packaging

The eight transducers have a beam width of about 10 degrees each (6dB down points) and are arranged with one looking upward, one downward, one looking to the left, one to the right, and four forward, as shown in Figure 2.2.2.

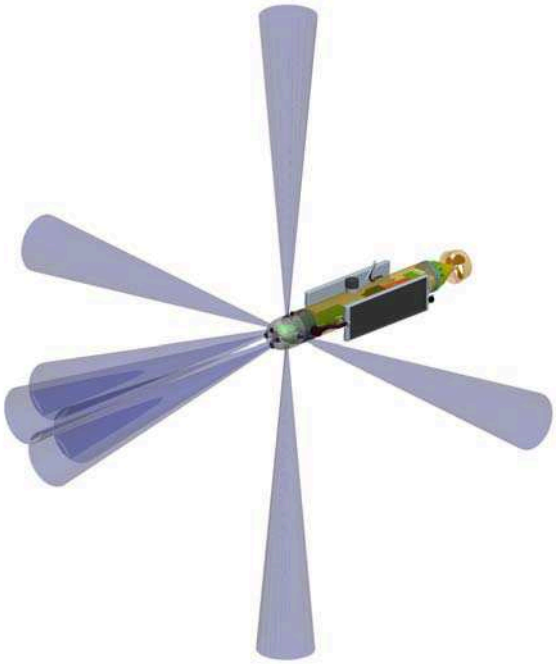


Figure 2.2.2: Obstacle Avoidance Sonar Beam Angles and Directions

The obstacle avoidance sonar transducers are all identical and interchangeable. Several of these transducers, seen in Figure 2.2.3, have been built and tested to determine their response and beam-width.

Figure 2.2.3; Obstacle Avoidance Sonar Transducers

A nose cone was fabricated in year two to accommodate this sonar package and is shown in Figure 2.2.4 along with the prototype processing electronics boards.

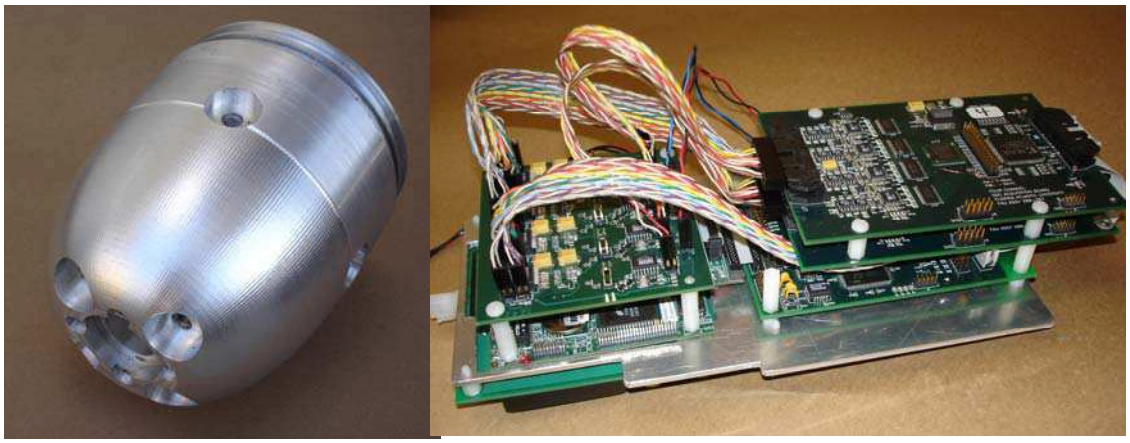


Figure 2.2.4: Obstacle Avoidance Sonar Nose-Cone and Electronics

The sonar was mounted on the vehicle during year three as shown in Figure 2.2.5. The nose cone also includes a window for an underwater video camera if required. The electronics of the sonar outputs the data from each transducer via a serial connector which is attached to one of the serial ports on the main computer in the vehicle. The software on the main computer reads the serial data and can be used to display the results in several different formats, as will be discussed in the next section. The data available gives the distance to obstacles in each of the eight directions shown in Figure 2.2.2.

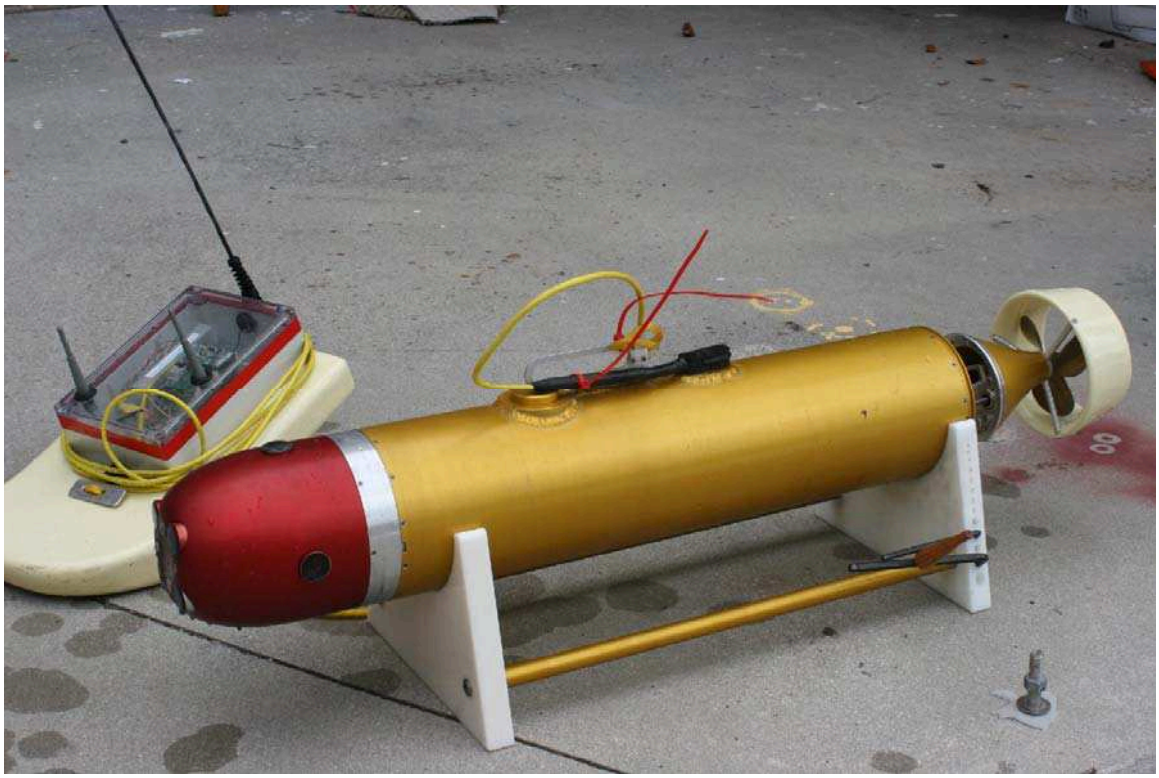


Figure 2.2.5: The obstacle avoidance sonar mounted to the RPUUV in tit's final form

2.2.3.2 Calibration of the Sonar

When the sonar had been mounted on the RPUUV a support system was built so the system could be calibrated in a test pool. The support held the vehicle in the horizontal plane and rotated the vehicle with a stepper motor. Data from the sonar transducers were collected at either 5 deg or 1 deg increments over one complete rotation of the vehicle. The response gave the distances from each transducer to the wall of the tank and provided a measure of transducer response at smaller grazing angles. Also mounted in the pool was a 10" x 10" flat plat or a 1.375" pole that provided additional targets. The results are shown in Figure 2.2.6 and 2.2.7, and the outline of the test tank is clearly discernable from these results

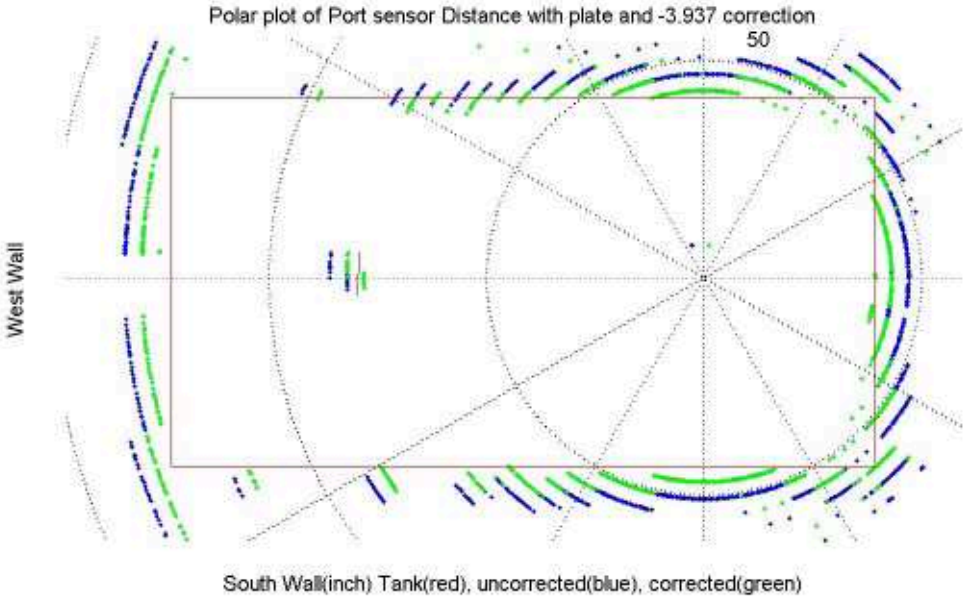


Figure 2.2.6 The calibration of the obstacle avoidance sonar showing the walls of the test tank. The green dots show the impact of introducing a 3.937 inch correction to the data. Note that a 10"x10" plate is mounted in front of the west wall and is correctly located.

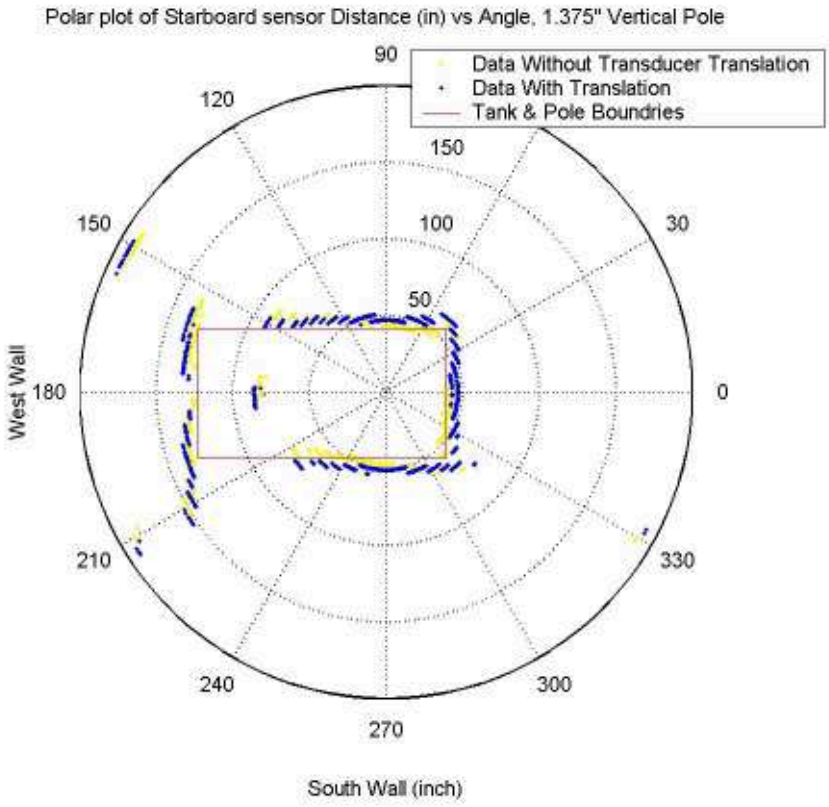


Figure 2.2.7 The calibration of the obstacle avoidance sonar showing the walls of the test tank. The green dots show the impact of introducing a 3.937 inch correction to the data. Note that a 1.375" pole is mounted in front of the west wall and is correctly located.

The conclusion from this test were that the sonar was accurate to within 3.93 " and identified specula reflections up to an angle of incidence of about 45 deg. It was also found that the plate could not be identified if it was rotated so the angle of incidence was 45 deg.

2.2.3.3 In Water Testing

In water testing took place in Whiskey Creek next to FAUs SeaTech complex (see for example Figures 2.2.15 and 2.3.5) and in the test pool on FAUs Boca Campus (Figure 2.2.8). The Whiskey Creek location provided an environment which is typical of a shallow water harbor, including the effect of seawalls, and the pool provided a very confined waters environment which tested vehicle control in restricted waters.

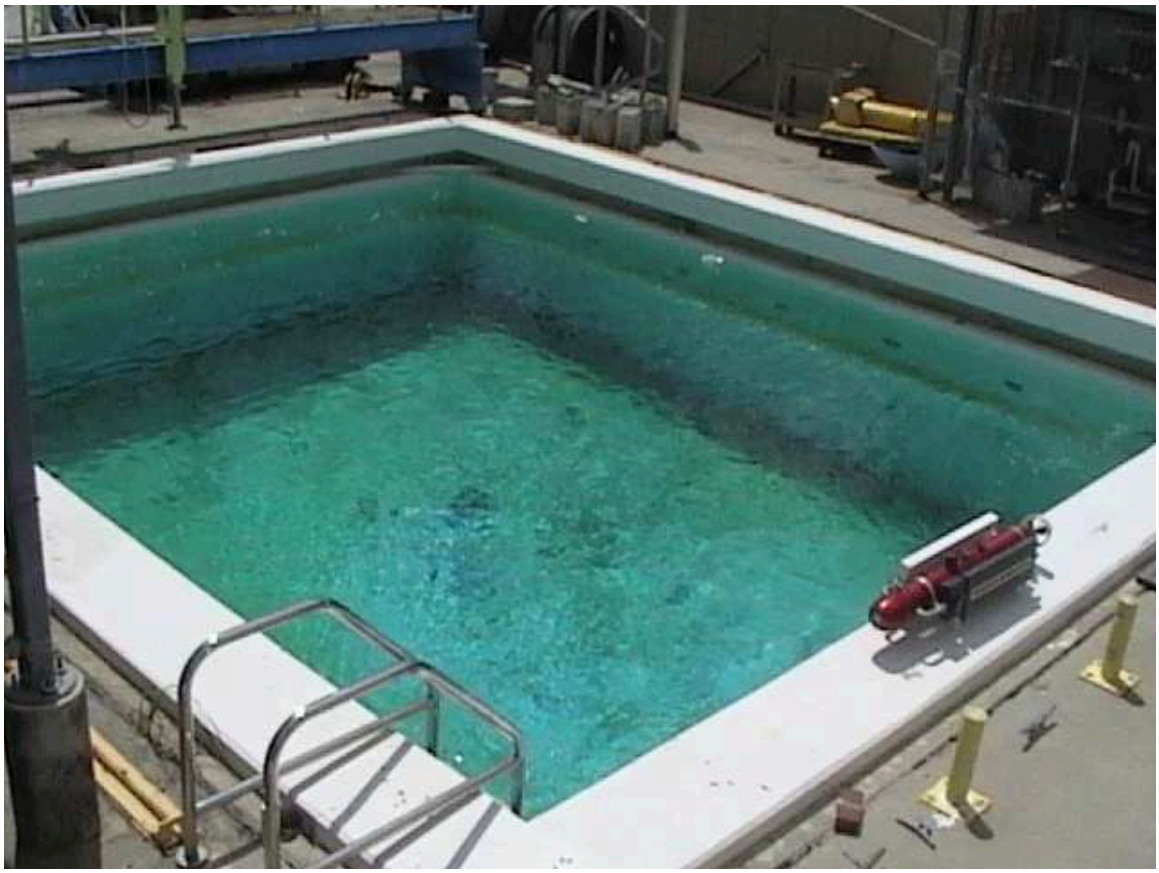


Figure 2.2.8: The test pool on FAUs Boca campus used for in water testing.

The vehicle was found to be readily controllable in both environments using either the RF link through a tow float or the acoustic modem providing that the vehicle was properly ballasted and supported by a tow float. The vehicle needed to be horizontal and negatively buoyant in a static condition so that the support to the tow float was taught as shown in Figure 2.2.9.

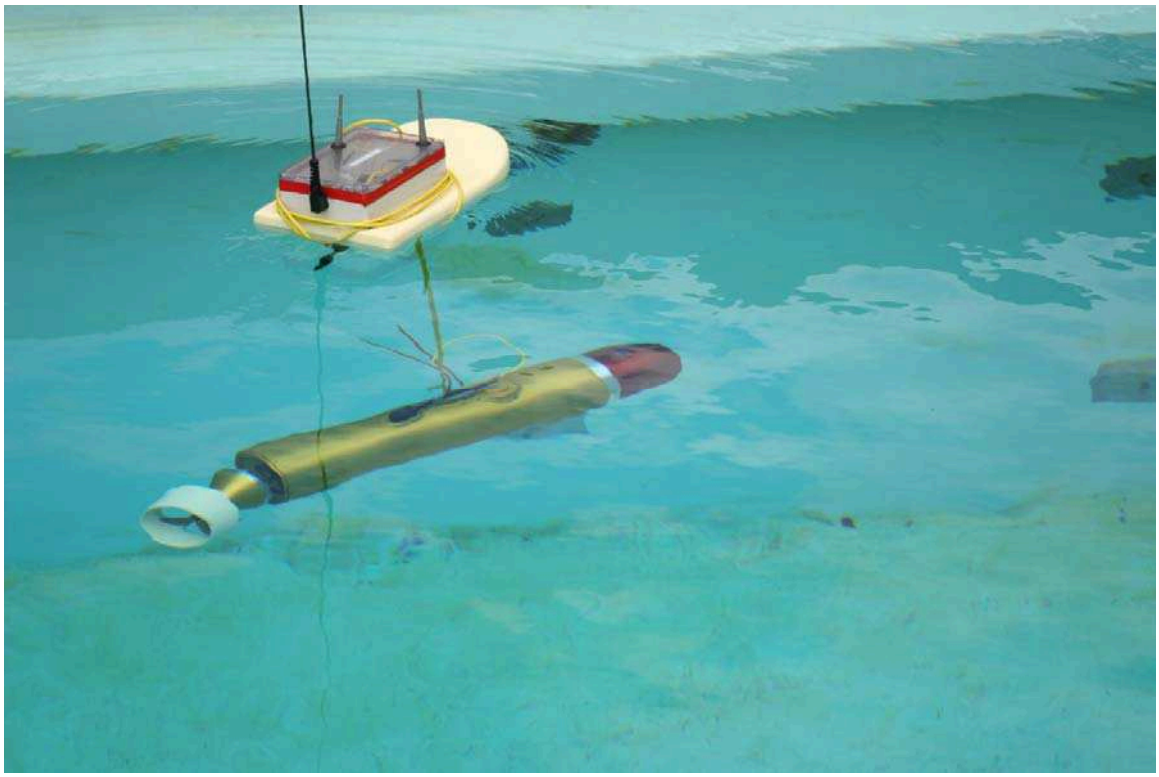


Figure 2.2.9 The vehicle operating in the test tank on FAUs Boca Campus

The obstacle avoidance sonar output was displayed on the topside console in various different formats. In the first instance the display used was as shown in figure 2.2.10. In the top left corner an XY plot of the vehicle position is presented showing the vehicle position. The bottom left corner shows the depth and altitude of the vehicle as a function of time, and the bottom right display shows the distance to obstacles as a function of polar angle, with the vertical direction being straight ahead. All displays were updated three times a second and the XY plot showed the movement of the vehicle. The XY display required the vehicle speed to be known, and while this worked well in simulations, in water testing showed that the display was misleading if the speed was inaccurate. An attempt was made to use the obstacle avoidance sonar to estimate the vehicle speed but this was very unreliable. Monitoring the propeller rpm was also not an effective measure of speed, especially during turns. This proved to be a limiting issue and so the XY plot was dropped from the display, and the simpler display shown in Figure 2.2.11 was used.

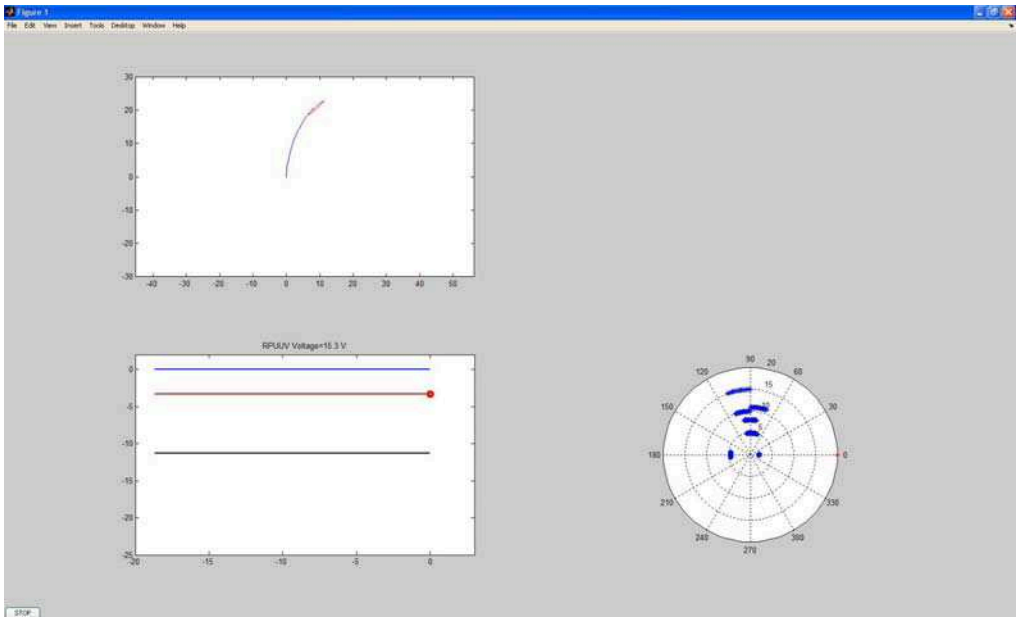


Figure 2.2.10: The topside display. In the top left corner an XY plot of the vehicle position is presented showing the vehicle position. The bottom left corner shows the depth and altitude of the vehicle as a function of time, and the bottom right display shows the distance to obstacles as a function of polar angle, with the vertical direction being straight ahead.

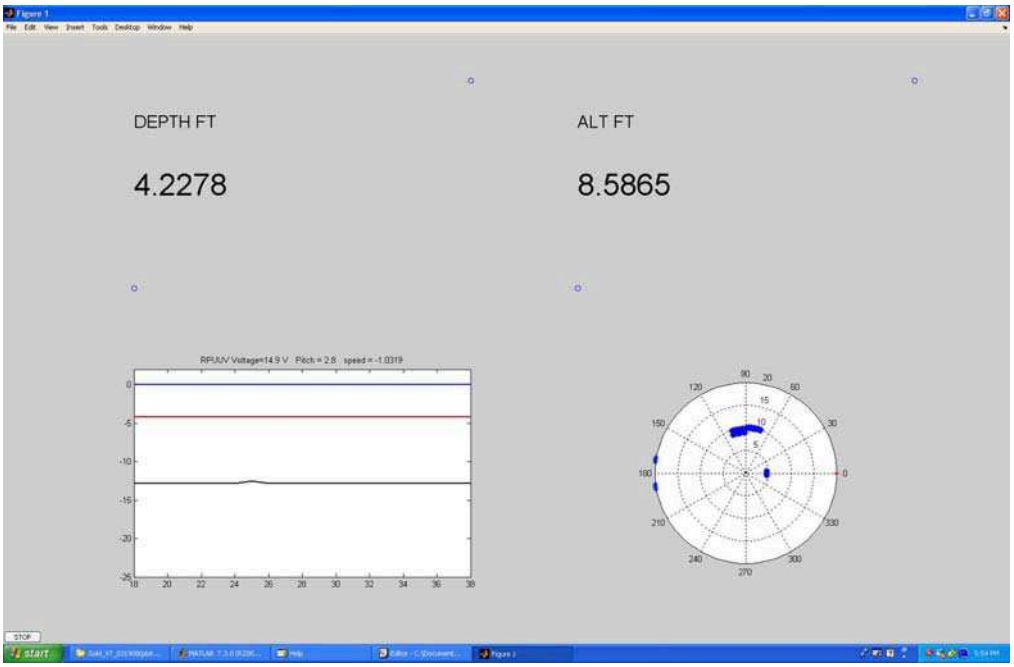


Figure 2.2.11: The modified topside display. The bottom left corner shows the depth and altitude of the vehicle as a function of time, and the bottom right display shows the distance to obstacles as a function of polar angle, with the vertical direction being straight ahead. The depth and altitude are also displayed numerically.

Using the modified display a test was carried out in the Boca test pool and it was found that an operator could navigate around in the restricted waters, even if he was unsighted and unaware of the vehicle position, by using the display output alone. It was therefore concluded that the simpler display was more effective for vehicle control than the more sophisticated versions because of the inaccuracy associated with speed determination. Other types of vehicles, such as AUVs, are able to navigate effectively because they have on board speed sensors which are very accurate. However the inertial navigation systems or Acoustic Doppler devices required to measure the speed of an underwater vehicle are high expense items. The objective of developing this vehicle was to minimize the end cost, and this was achieved by eliminating the need to install a sophisticated navigation system. The results shown in the test described above show that for this vehicle, operations using a tow float are very straight forward without additional navigation capabilities, and, for many of the applications envisioned for the RPUUV, operations with a tow float meet all requirements

.

2.2.4 Fabrication, Installation & Testing of the Chemical Sensor Payload

In year two of this work, a method for the in-situ detection of explosive chemicals was developed by Dr. Richard Granata. The details of this method are presented in section 2.2.8 of this report. In this section we describe how this methodology was implemented into a compact, stand-alone payload suitable for installation on the RPUUV.

The explosives detection is essentially achieved by pumping in the ambient seawater and mixing it with an on-board chemical reagent. The reagent has a fluorescence that is quenched by the seawater unless explosive trace elements are present. Therefore, by passing the mixture through a modified underwater fluorometer it is possible to detect the increase in fluorescence that occurs in the presence of explosive trace elements such as nitroglycerine.

2.2.4.1 Design & Installation of the Chemical Sensor Payload

A schematic of the explosives detection payload can be seen in figure 2.2.12. The system consists of two micro-pumps; one that pumps in seawater at about 75cl/min, and another that pumps the reagent from two bladder style accumulators at a rate of about 7.5cl/min. The seawater and reagent are then mixed together in a short (6”) static mixing tube before passing through an 8ft coil of tubing to give the reaction time to develop. After about 1 minute the mixture then passes through the fluorometer, where any explosives are detected, before being expelled back to the surrounding environment. To ensure that the net buoyancy of the payload does not change as the reagent is consumed, the void left behind the accumulator bladders as they are depleted is allowed to fill with the venting mixture.

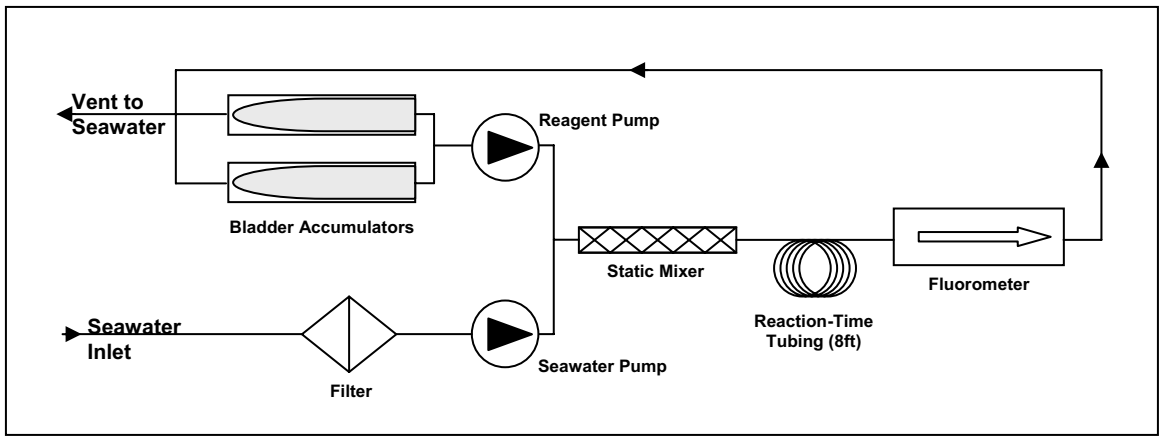


Figure 2.2.12: Chemical Sensor Payload Schematic

CAD models were developed of all the major components of the payload to facilitate their packaging into as small a volume as possible within the 6"OD of the RPUUV form-factor. Since the nature of this package involved pumping fluids into and out of an otherwise dry vehicle, to protect the batteries and electronics of the vehicle from possible leaks, it was decided to isolate this payload in its own pressure housing that would mount directly to the front of the RPUUV via a double sided endcap. A view of the Pro-Engineering solid model, showing the relative placement of the major components, is presented in figure 2.2.13.

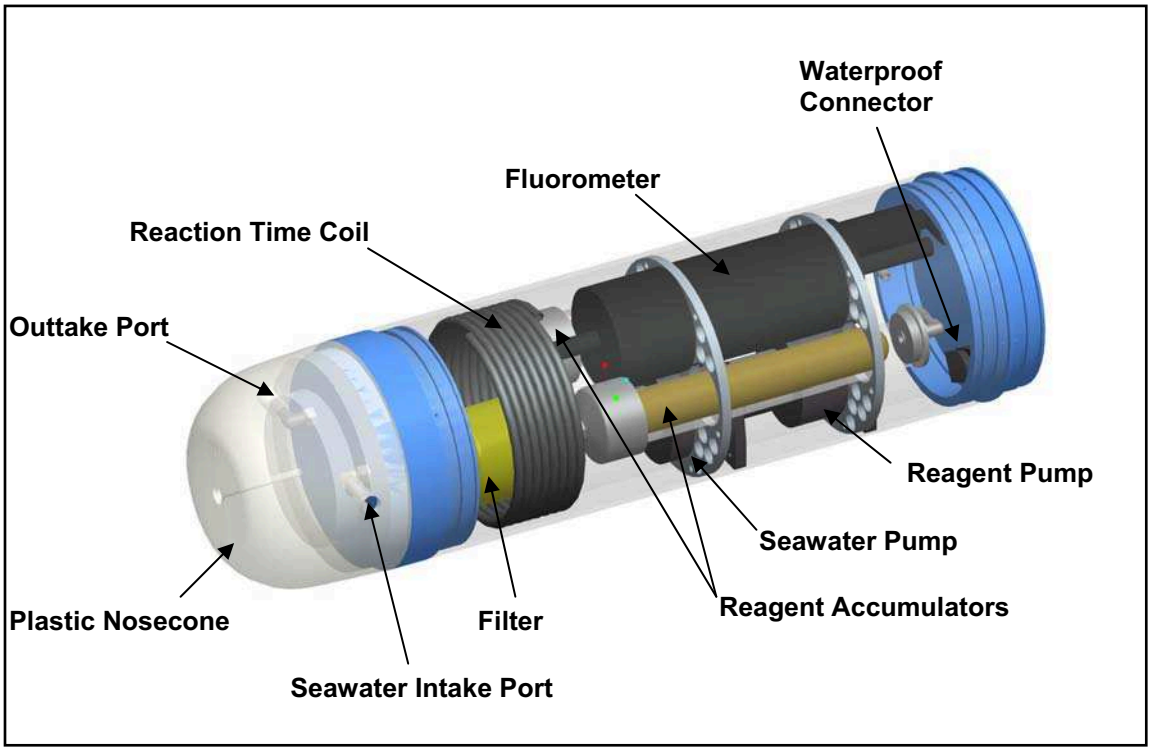


Figure 2.2.13: CAD Model of Chemical Sensor Component Packaging

Using a waterproof 8-pin bulkhead connector it was possible to place the pump control and power circuitry in the main pressure housing of the RPUUV and pass the serial output from the fluorometer directly to an input on the vehicles main PC104 computer.

Figure 2.2.14 shows the actual fabricated payload with the same components as the CAD model. In these photographs it is possible to see some of the additional hoses, fittings and electrical connections that are necessary to complete the fluid and electrical circuits inside the payload. Outside of the payload pressure vessel we can see the pump speed control potentiometer circuits and the 12-24V DC-DC converter required to supply 24Vdc to the pumps. The db-9 serial interface connector that plugs into the vehicle main processor is also visible. The bottom picture in figure 2.2.14 shows the chemical sensor payload fully integrated onto the nose of the RPUUV

Figure 2.2.14: Packaged Chemical Sensor Payload & RPUUV Mounting

2.2.4.2 In-Water Testing of the RPUUV with the Chemical Sensor Payload

Laboratory bench-top testing was performed to tune the pump delivery rates and reagent concentrations and to calibrate the fluorometer. Details of these tests are presented in section 2.2.8. In these tests, small quantities of medical grade nitroglycerine pills were dissolved in containers of seawater to act as a positive detection media. It was considered infeasible and environmentally damaging to prepare sufficient nitroglycerine infused water to do any real explosives detection in the port/marina environment. However, although no explosives were detected, the successful integration and operation of the RPUUV outfitted with the explosives detection payload was demonstrated in the creek adjacent to Florida Atlantic University's SeaTech campus, and pictures from these trials are shown in figures 2.2.15 and 2.2.16.

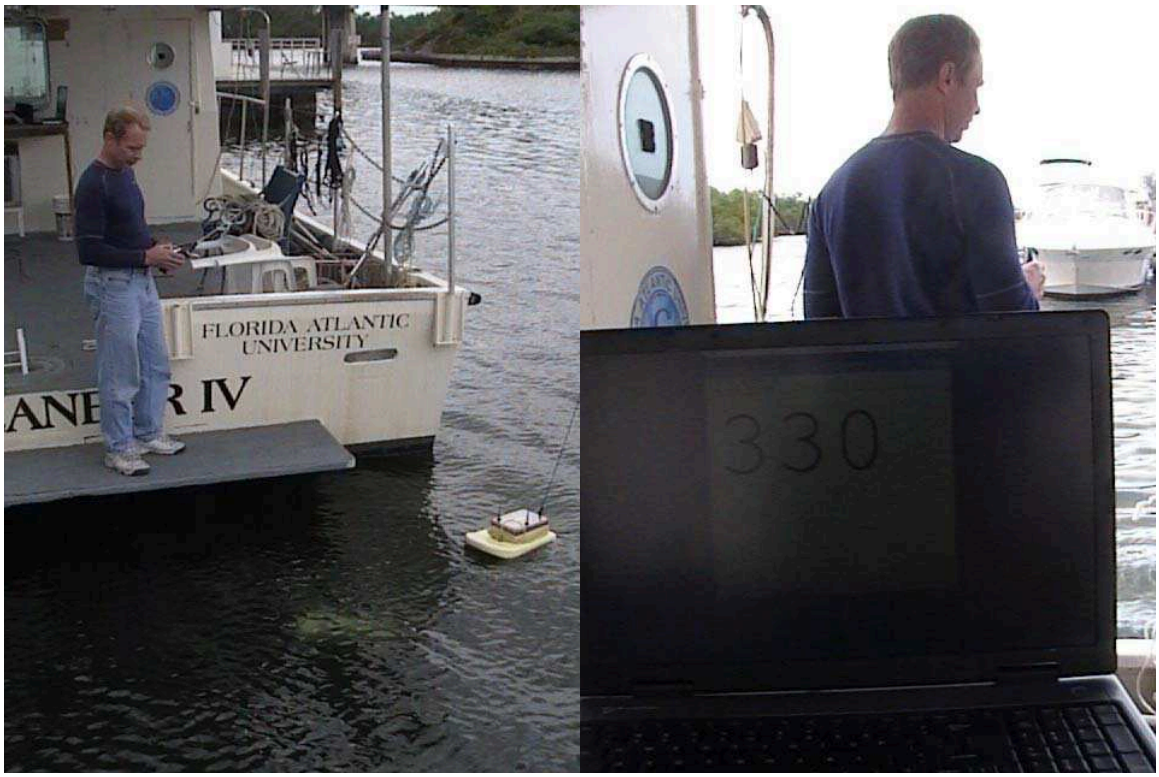


Figure 2.2.15: Operating the RPUUV with the Chemical Sensor Payload

In figure 2.2.15 the operator is seen using the RC controller to maneuver the RPUUV adjacent to a small research vessel in the confined quarters of a small marina. The output from the sensor payload's fluorometer is relayed to the topside computer via the tow-float RF link and can be seen displayed on the laptop computer in this figure.

Figure 2.2.16: In-Water Demonstration of the RPUUV with the Chemical Sensor Payload

2.2.5 Mounting and Testing of High Resolution Sonar on the RPUUV

In years one and two of this work, Dr. Steven Schock of Florida Atlantic University, has been developing a very high resolution sonar system for integration as a payload onto the RPUUV. Complete details of this sonar development, along with some initial test images are presented in section 2.5 of this report. In this section we describe how the RPUUV was modified to accommodate the sonar arrays, its acquisition and image processing electronics, and the additional power source requirements associated with this high resolution sonar package. Initial in-water testing of the integrated system is also documented.

2.2.5.1 Packaging & Installation of the High Resolution Sonar Payload

A 3-dimensional CAD model of the sonar processing boards, CPU card and additional battery set are presented in figure 2.2.17. Also seen in this figure is the adapter mounting ring that was designed so that the sonar arrays could be relatively easily added and removed from the vehicle. This adapter ring also provides a mechanical location for the cable penetrations that are required to pass the large bundle of wires from the sonar arrays to the processing hardware inside the main pressure vessel. The processing cards and CPU are mounted on a chassis above a set of batteries similar to those in the main RPUUV section. In this way the whole sonar system is essentially independent of the rest of the vehicle. An Ethernet connection from the sonar system to the hub in the vehicle provides the user with the same real-time interface with the sonar CPU via the wireless bridge on the tow-float.

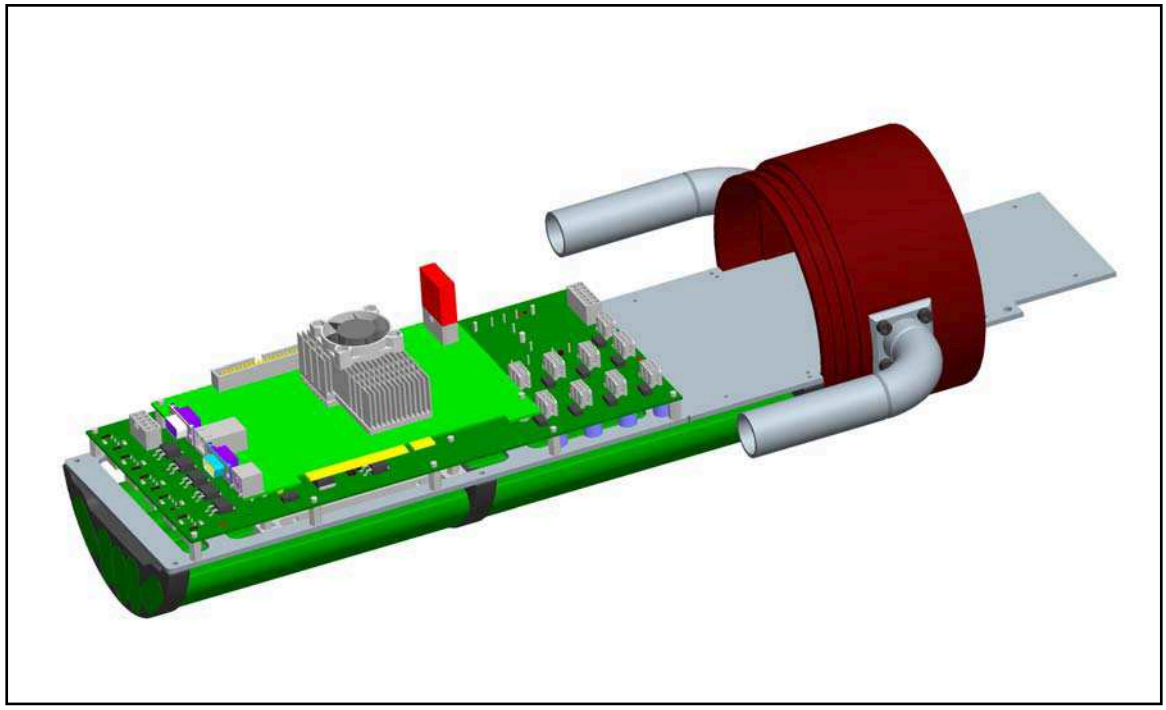


Figure 2.2.17: High Resolution Sonar Electronics, Batteries, & Mounting Adapter Ring

In figure 2.2.18 it can be seen how this configuration preserves the internal space requirements of the obstacle avoidance sonar electronics and nose section (described in section 2.5), so that both systems can be integrated and operated simultaneously.

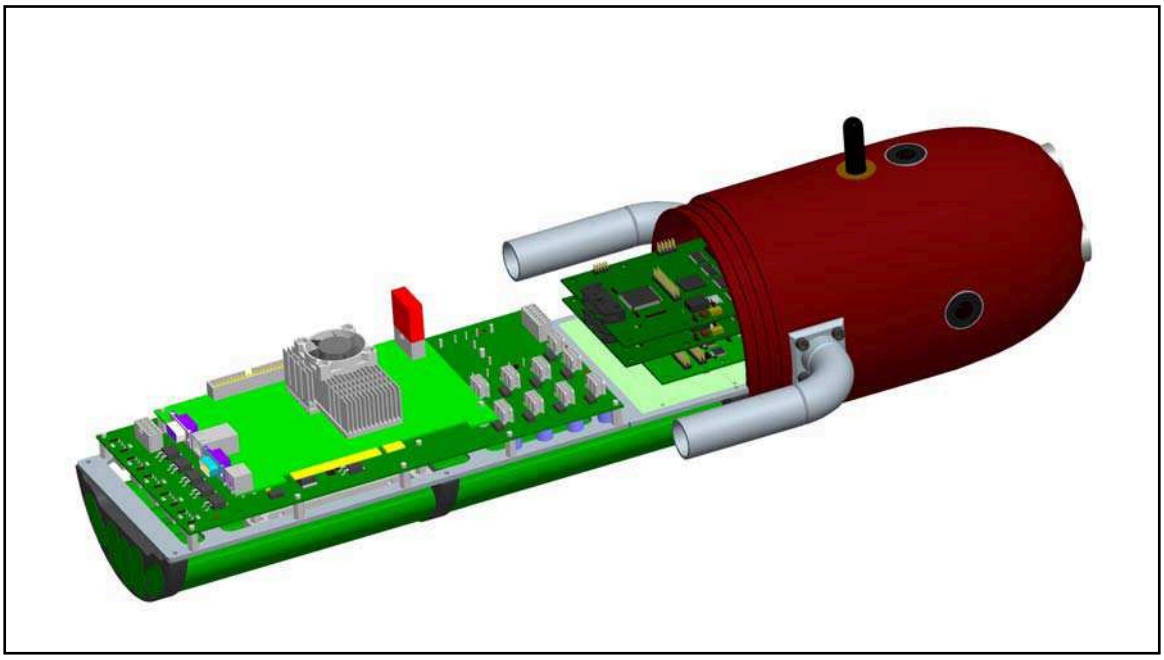


Figure 2.2.18: Packaging of the High Resolution & Obstacle Avoidance Sonar

The arrays of the high resolution sonar each contain 512 elements arranged in a straight line about 30" long. These elements, along with their preamplification circuit boards are housed in oil-filled panels that are mounted to the outside of the RPUUV parallel to the mid-section of the vehicle. The sonar projector hydrophone and power amplifier are similarly packaged and attached in line with the array. The mounting arrangement for these panels is presented in figure 2.2.19. Shown in this figure are the two mounting rings that are clamped around the outside to the RPUUV's parallel mid-section. The array supports are then clamped into these rings securing the panels to the sides of the vehicle and allowing their angle to be adjusted or pivoted so that the sonar can look in an upward or downward direction. The pivot mounts on the arrays have o-ring seals that mate with the cable penetrator tubing allowing the arrays to be pivoted while still retaining a water tight connection.

Figure 2.2.20 shows the fully assembled vehicle with both high resolution and obstacle avoidance payloads installed.

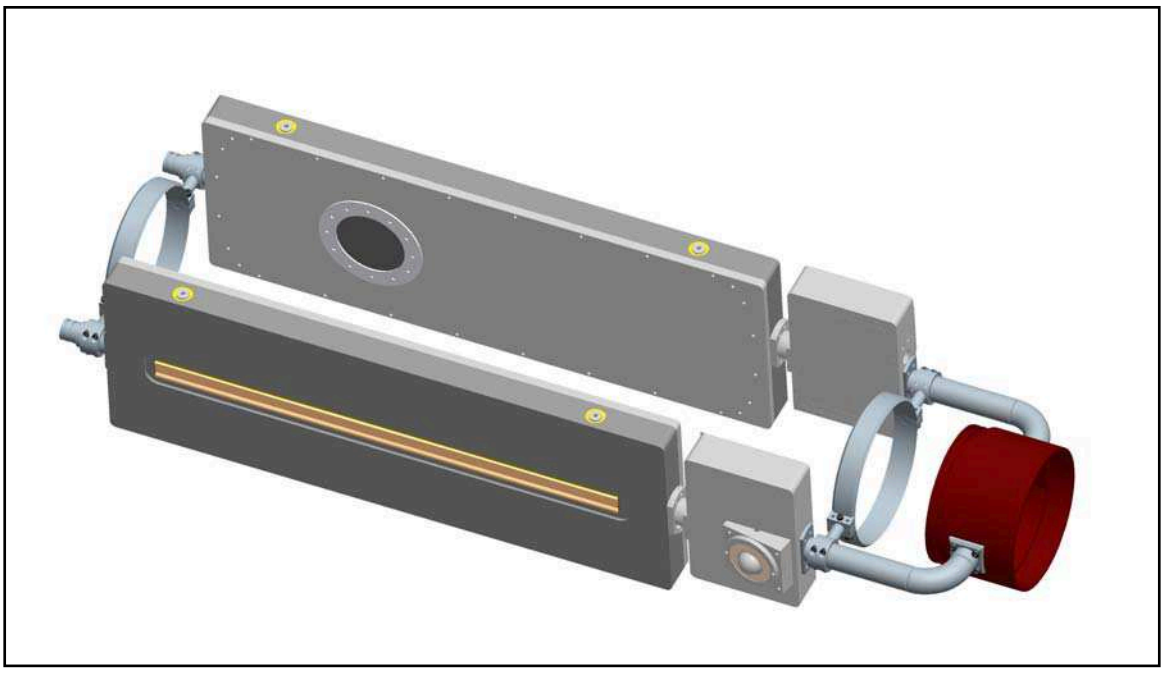


Figure 2.2.19: High Resolution Sonar Array Panels, Cable Penetrations & Mounting Hardware

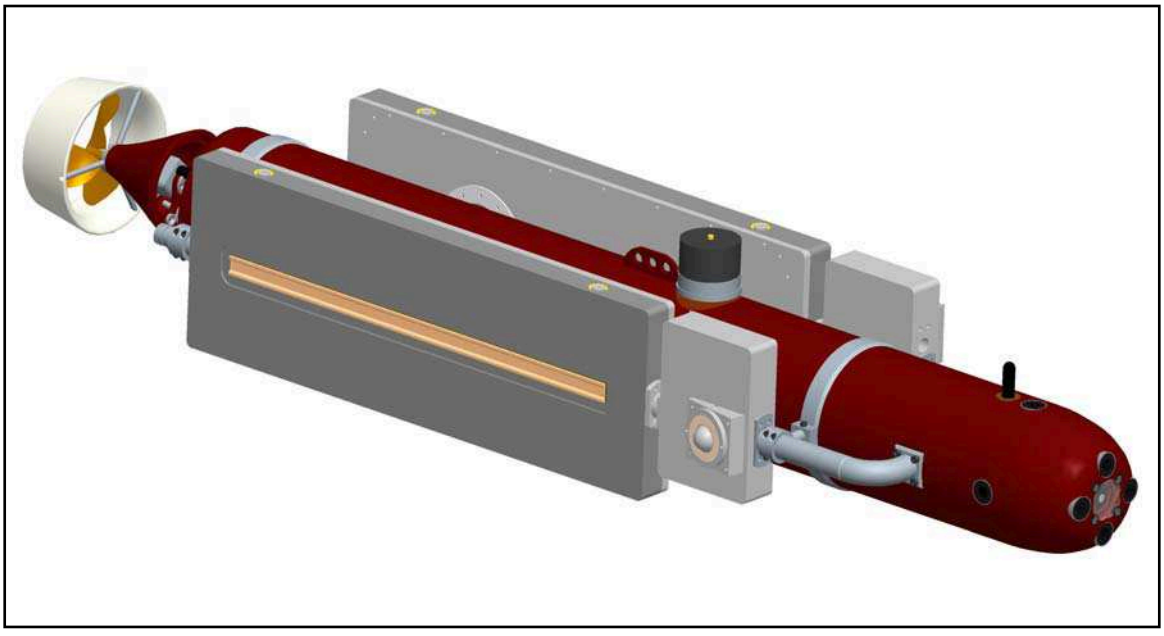


Figure 2.2.20: High Resolution & Obstacle Avoidance Sonars Mounted on the RPUUV

2.2.5.2 In-Water Testing of the RPUUV with the High Resolution Sonar Payload

Only one initial prototype array panel has actually been constructed for testing. A photograph of this array can be seen attached to the RPUUV in figure 2.2.21. To balance the vehicle in both roll and hydrodynamic drag, a dummy array panel was constructed and attached to the starboard side of the RPUUV.

Figure 2.2.21: The RPUUV with the High Resolution Sonar Payload

The RPUUV was fitted with the high resolution sonar package and tested in the marina adjacent to Florida Atlantic University's SeaTech campus. The vehicle, with the sonar, was then used to image items on the marina bed and then maneuvered back and forth along the side of the 65' aluminum hulled vessel RV Stephan to image the underside around the boats running gear. Acoustic images of these items as well as images of the nearby bridge support pilings are presented in section 2.5 of this report along with photographs from the test site.

2.3 Acoustic Communications

PI: Dr Pierre Beaujean

Tasks 3.6-3.8

2.3.1 Summary

The main objective of this portion of the project is to achieve communications for the purpose of transmitting and receiving information wirelessly between a user and the Remotely Piloted Underwater Vehicle (RPUV). Transmitted information is used to pilot the RPUV and relay its position. Information received from the RPUV combine acoustic images of the environment and status report of the vehicle. During the first year of this project radio wave (WiFi) communication was used to control the vehicle. Whenever the tow-float solution becomes impractical, a slower but fully wireless acoustic modem is to be used. The design must consider the issues associated with acoustic communications in port at high data rates, using a high-frequency acoustic modem, and the piloting and tracking of the RPUV, using a command-and-control acoustic modem.

The objectives for year 3 were the experimentation and fine tuning of the piloting (command-and-control) acoustic modem, the acoustic positioning unit and the high-frequency acoustic modem for image transmission. All the objectives have been completed:

- The piloting acoustic modem has been tested at the SeaTech marina at a maximum range of 75 m. The vehicle moved at a top speed of 0.5 m/s in 0.5 to 3 m of water. Acoustic propagation study in a port environment was also completed to aid the analysis and prediction of performance of the underwater acoustic piloting system. The model was been tested against physical measurements in the south turning notch of Port Everglades, Florida. The impulse response could be modeled with a relative echo magnitude error of 1.62 dB at worst, and a relative echo location error varying between 0% and 4% when averaged across multiple measurements and sensor locations.
- The acoustic positioning unit has been tested in a calibration tank and at the SeaTech marina at Florida Atlantic University, Dania Beach. In two meters of water and in the presence of multiple walls and boats, the estimation error in source azimuth is 0.9 degree at 20 meters.
- The high-frequency acoustic modem was installed in the remotely-piloted vehicle and transmitted snippets of canned information to the HS-HFAM receiver. These acoustic images were received at a rate of 4 per second during a set of experiments in the FAU SeaTech marina. The vehicle moved at a maximum speed of 0.5 m/s. The RPUV was piloted acoustically during this set of experiments, and no loss of performance was noticed either in terms of low-frequency acoustic piloting or in terms of high-frequency acoustic data transmission. The maximum distance from between the vehicle and the high-frequency modem receiver was 60 m.

2.3.2 Introduction

Operational criteria:
The objective of this research is to be capable of piloting an underwater vehicle remotely using either radio frequencies or sound. This vehicle is to perform search missions, principally in ports and very shallow waters, to find potentially dangerous or illegal objects such as explosive or narcotics. Note that this vehicle can also perform scientific missions. In its initial configuration, the vehicle is equipped with:

- An imaging sonar system, a camera and an optional chemical sensor for threat detection.
- A tilt sensor and compass, and an Ultra-Short Baseline acoustic positioning system.
- An embedded processor, a motherboard and Ni-Mh batteries.
- A tow-float with a WiFi access point (802-11g, 2.4 GHz) for image and data transmission and Radio Frequency (72 MHz) control unit for piloting.

In a second, fully untethered configuration, the vehicle is equipped with the acoustic communication package:

- A low-speed acoustic modem for remote piloting and positioning (surface to vehicle).
- A high-speed acoustic modem for image transmission (vehicle to surface).

Acoustic remote piloting and positioning:
The criteria retained for the piloting and positioning of the RPUV are as follows:

- The vehicle is to operate for approximately 2 hours in approximately 1 to 20 m of water, in the proximity of walls, pilings and underneath ships.
- The vehicle is moving at a top-speed of 1 m/s, at a maximum range of 100 m.
- The vehicle is assumed to remain at least 0.25 m from the surface during operations.
- The peak power consumption of the modem receiver unit in the vehicle is to be kept to a minimum (0.5 W) and use as few transducers as possible (one ITC-3460 and one ITC-1089D).
- At the surface, the source level is limited to 168 dB re 1μPa//1m averaged over time, due to environmental requirements, and must not interfere with the imaging sonar nor the USBL.
- The remote piloting modem must operate so that it can easily replace the remote piloting unit initially used for piloting.
- The pitch, yaw and thrust of the vector thruster unit must be updated at least 3 times every two seconds, with 128 positions for pitch and yaw, and 128 levels of thrust.

High-speed acoustic communications:
The criteria retained for high-speed acoustic communication system, used to transfer video and sonar information, are as follow:

- A maximum achievable data rate of 87,768 bits per second at fairly close range (150 meters and less) in harbors and in very shallow water.
- Small, low-power and inexpensive device, well suited for modern untethered underwater vehicles operating in very shallow water and ports.
- Compressed video or high-resolution sonar images should be relayed to a topside unit in real-time.

2.3.3 Acoustic remote piloting and positioning:

2.3.3.1 System overview:

The objective is to pilot the vehicle using sound. In this configuration, the RPUV does not use a tow-float. Instead, two acoustic communication units are used for piloting and navigation, and to relay video and images. An overview of the acoustically-piloted RPUV is shown in Figures 2.3.1 and 2.3.2.

Using acoustic communications to send command and control information to a UUV is not a unique concept [2-5], though previous attempts at remote piloting have not taken into consideration the concept of real-time command and control. The technique used most often is supervisory control instead of true joystick-type remote control. This sort of control is used mostly to send new autopilot algorithms or preprogrammed sequences from the pilot to the UUV. One example of true joystick-type remote control is given in [5], where the vehicle is piloted once on the surface of the water using a wireless RS-232 uplink, to aid in the launch and recovery of the vehicle and is only functional on the surface. The concept of full joystick-type command and control of a UUV using acoustic communications is a novel concept that is being developed at the Center for Coastline Security and Technology [1][17].

The piloting and positioning unit uses an FAU-Dual Purpose Acoustic Modem (DPAM) [6][7][20], which transmits remote-piloting commands from the topside to the underwater vehicle and receives position information back from the vehicle periodically. The topside unit is equipped with an ITC-3460 reciprocal transducer for piloting. Acoustic positioning takes place using a top-side FAU Ultra-Short Baseline (USBL) array [8][9][21]. A picture of the FAU DPAM electronics is shown in Figure 2.3.3.

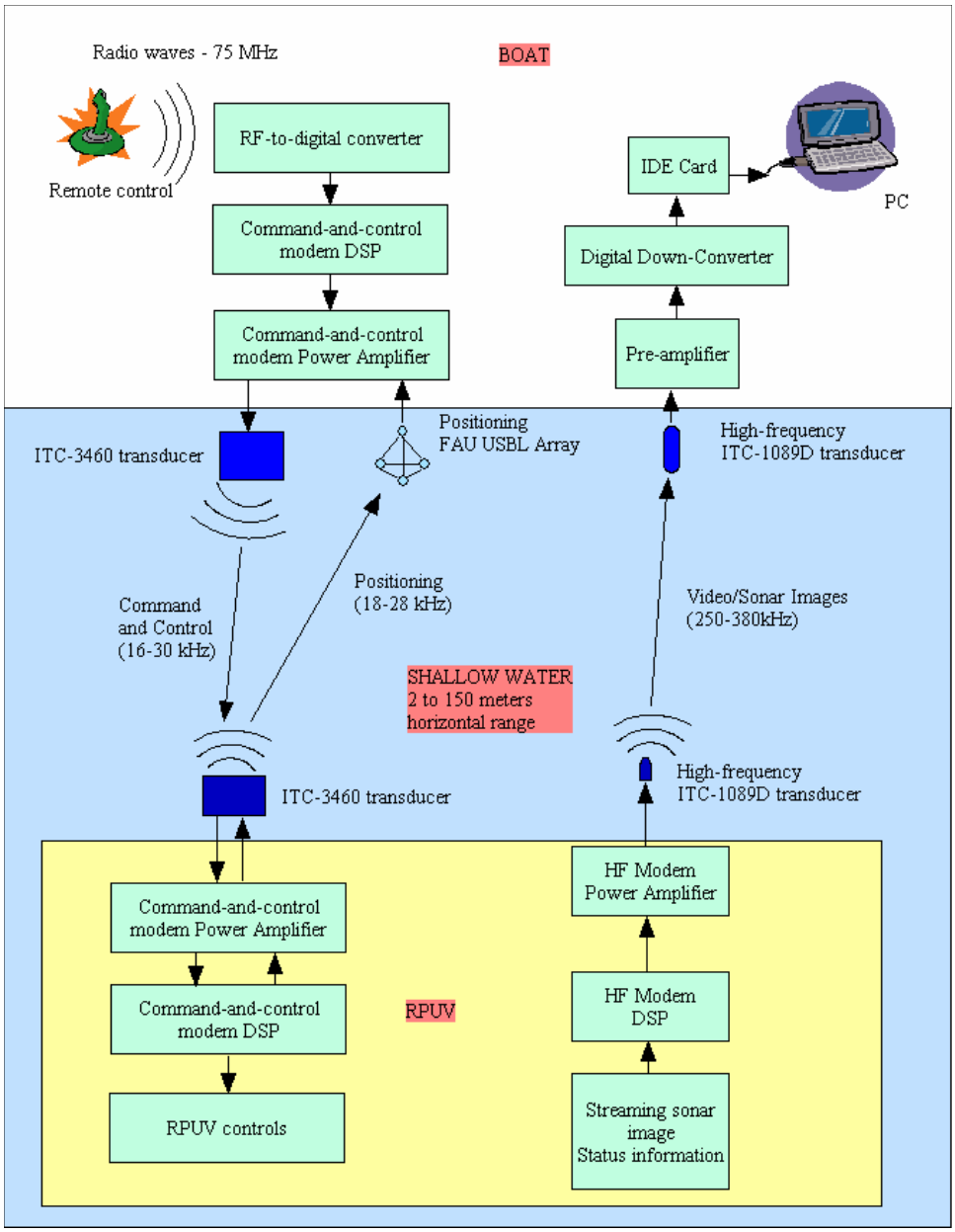


Figure 2.3.1. Overview of the RPUV control using a tow-float and acoustic waves.

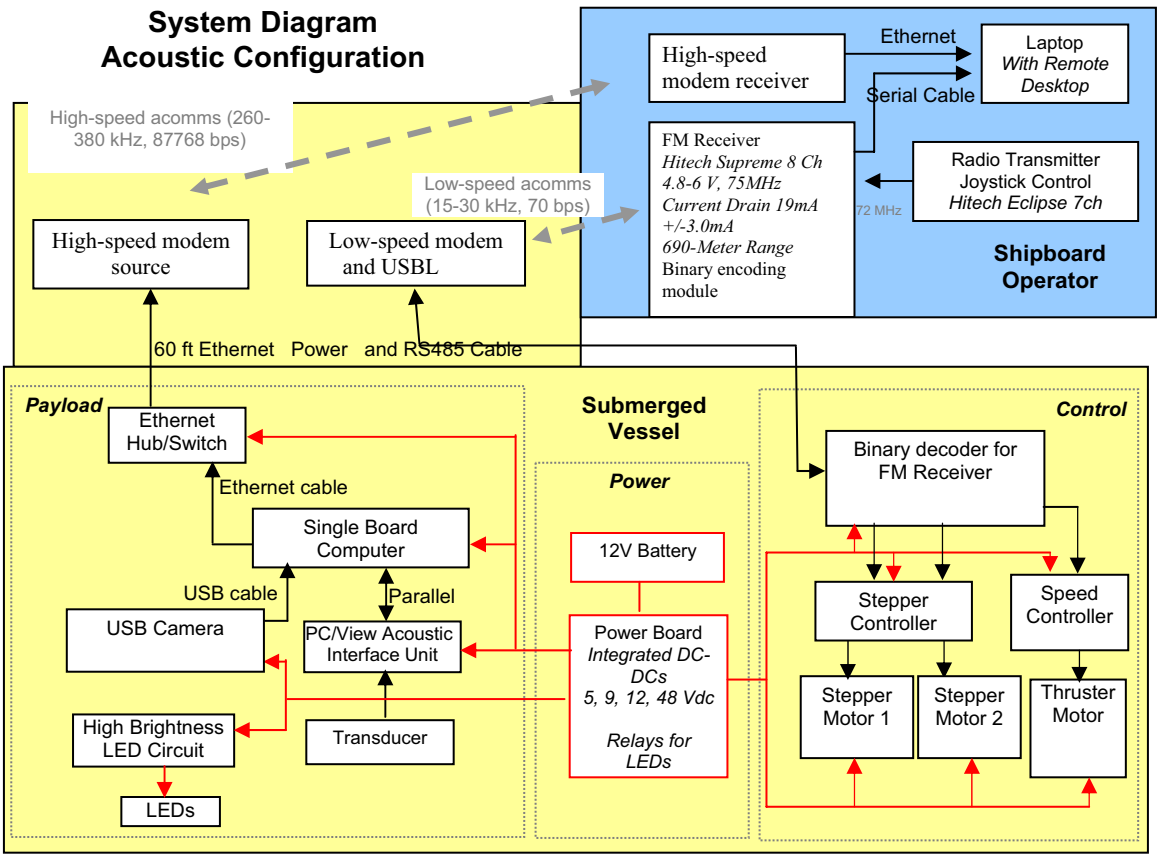


Figure 2.3.2. Detailed diagram of the RPUV control using acoustic waves.

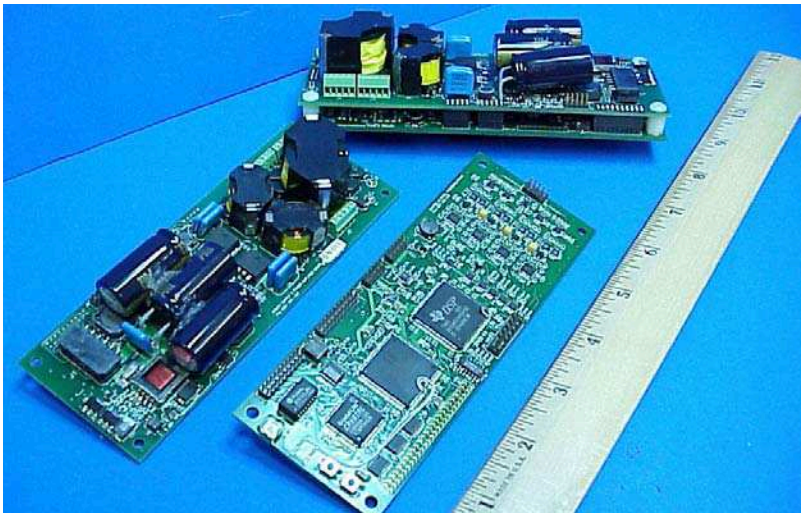


Figure 2.3.3. Acoustic remote piloting electronics.

At present, the remote piloting unit is capable of transmitting up to 3 piloting messages every two seconds, while the vehicle position is updated approximately once every 2 seconds. The tested range for piloting is approximately 30 m, with an estimated maximum range of 3000 meters at full power based on previous experimentation of the FAU DPAM. The electronic units are built, and the RPUV is already equipped with an

FAU DPAM electronic DSP card and amplifier, and an ITC-3460 transducer. The FAU USBL unit has also been assembled. The remote piloting software source and receiver has been completed and tested.

2.3.3.2 Remote acoustic piloting processing and experimentation:

Figure 2.3.4 provides the details of the remote piloting signal processing and hardware platform.

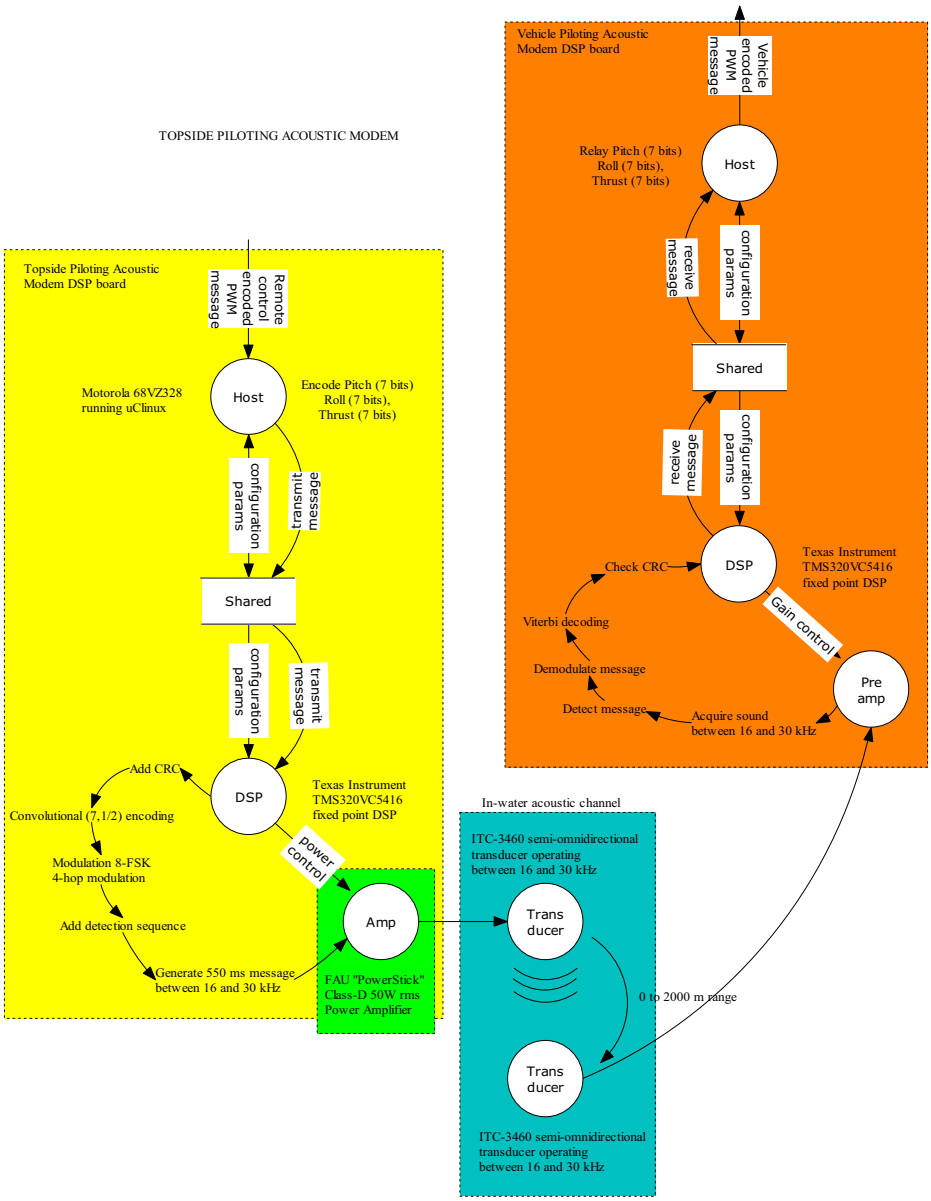


Figure 2.3.4. Detailed system diagram of the acoustic piloting software and hardware.

The acoustic piloting unit was tested on the RPUV was tested in the marina at the FAU SeaTech campus in Fort Lauderdale, Florida following the sequence shown in Figure 2.3.5 in the location shown in Figure 2.3.6. Details on this test can be found in [17].

Figure 2.3.5. Acoustic piloting of the RPUV: (top left) deployment, (top middle) close-up of vehicle and buoyancy float, (top right) acoustic piloting in SeaTech marina from the pilot view, (bottom left) acoustic piloting in SeaTech marina from a different angle, (bottom middle) acoustic piloting in the canal, (bottom right) recovery of the vehicle.

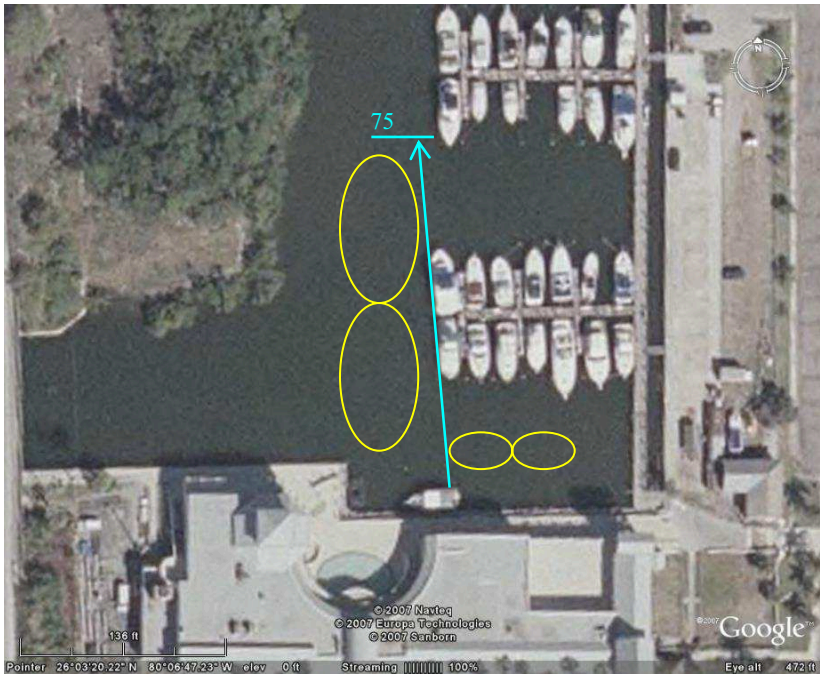


Figure 2.3.6. Aerial view of the FAU SeaTech marina.

Acoustic propagation study in a port environment was also completed to aid the analysis and prediction of performance of the underwater acoustic piloting system. To do so, a computer-efficient model for underwater acoustic propagation in a shallow, three-dimensional rectangular duct closed at one end has also been developed using the method of images [20]. The duct simulates a turning basin located in a port, surrounded with concrete walls and filled with sea water. The channel bottom is composed of silt. The

modeled impulse response is compared with the impulse response measured between 15 kHz and 33 kHz. The model has been tested against physical measurements in the south turning notch of Port Everglades, Florida. Figure 2.3.7 provides a summary of the experiment. Despite small sensor-position inaccuracies and an approximated duct geometry, the impulse response can be modeled with a relative echo magnitude error of 1.62 dB at worst, and a relative echo location error varying between 0% and 4% when averaged across multiple measurements and sensor locations. This is a sufficient level of accuracy for the simulation of an acoustic communication system operating in the same frequency band and in shallow waters, as time fluctuations in echo magnitude commonly reach 10 dB in this type of environment.

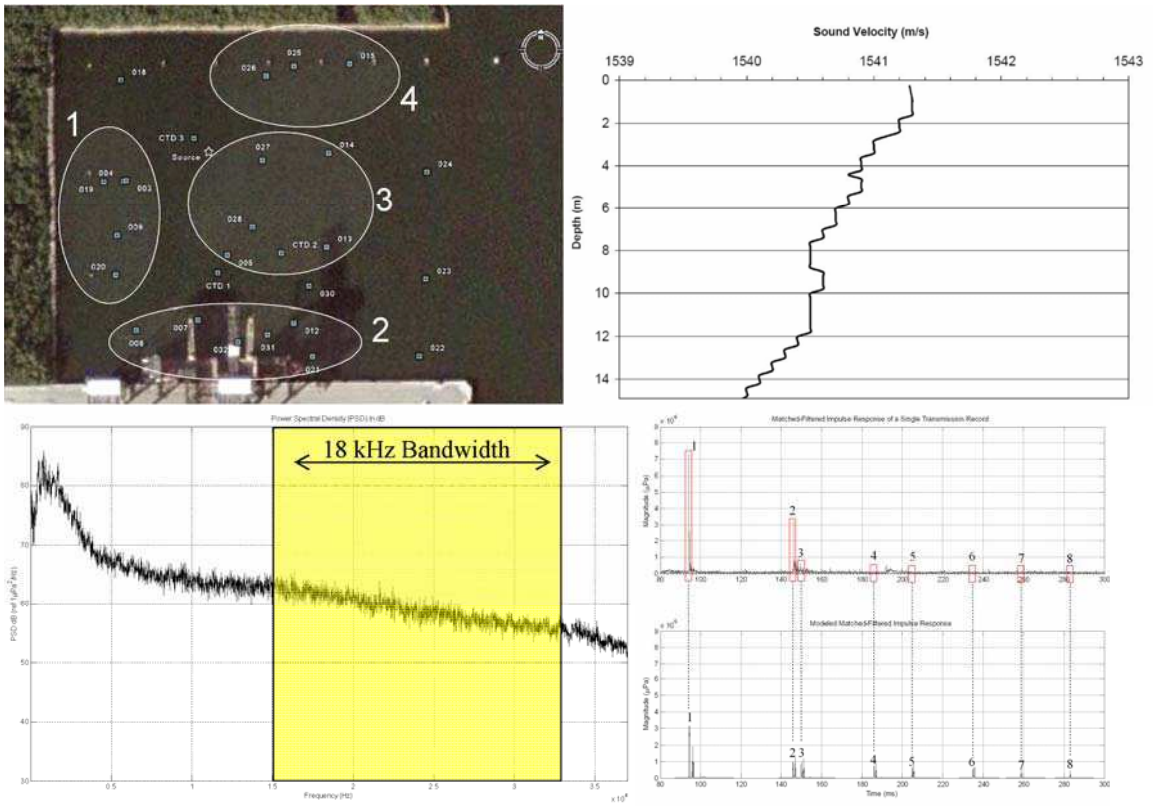


Figure 2.3.7. Acoustic characteristics of the south turning notch in Port Everglades, Florida: (top left) aerial view and measurement spots, (top right) sound velocity profile, (bottom left) ambient noise power spectral density, (bottom right) example of measured and simulated impulse responses.

2.3.3.3 USBL processing and experimentation:

This section contains the results obtained from a fully Motion Compensated (MC) USBL-APS implemented using the same signaling, range and angle-of-arrival estimation technique. The motion-compensation system estimates the array position and orientation while merging noisy measurements from a Magnetic, Angular Rate, and Gravity (MARG) sensor and a Differential Global Positioning System (DGPS), using Kalman filtering [19]. This APS can operate in volumes of water of less than 10 cubic meters, making it suitable for ports and very shallow-water operations. The system has been

tested in a calibration tank and at the SeaTech marina at Florida Atlantic University, Dania Beach. In two meters of water and in the presence of multiple walls and boats, the estimation error in source azimuth is 0.9 degree at 20 meters.
The Ultra-Short Baseline (USBL) Acoustic Positioning System (APS) is mounted on a surface vessel, which implies that the position of the tracked vehicle is expressed in the frame of the ship, or navigational frame. This position is actually not exploitable as the ship cannot keep a fixed position at the surface of the sea. The boat moves following the motion of the waves, the wind and also the current, or more simply because the crew wants to move the boat. Software has been developed to transform the USBL APS measurements in the north east down frame (NED frame) using an XSens MTi IMU, capable of sensing the motion of the platform. Figure 2.3.8 shows an overview of the USBL-APS unit. Figure 2.3.9 shows the USBL sensors. Figure 2.3.10 depicts a typical experiment setup and results.

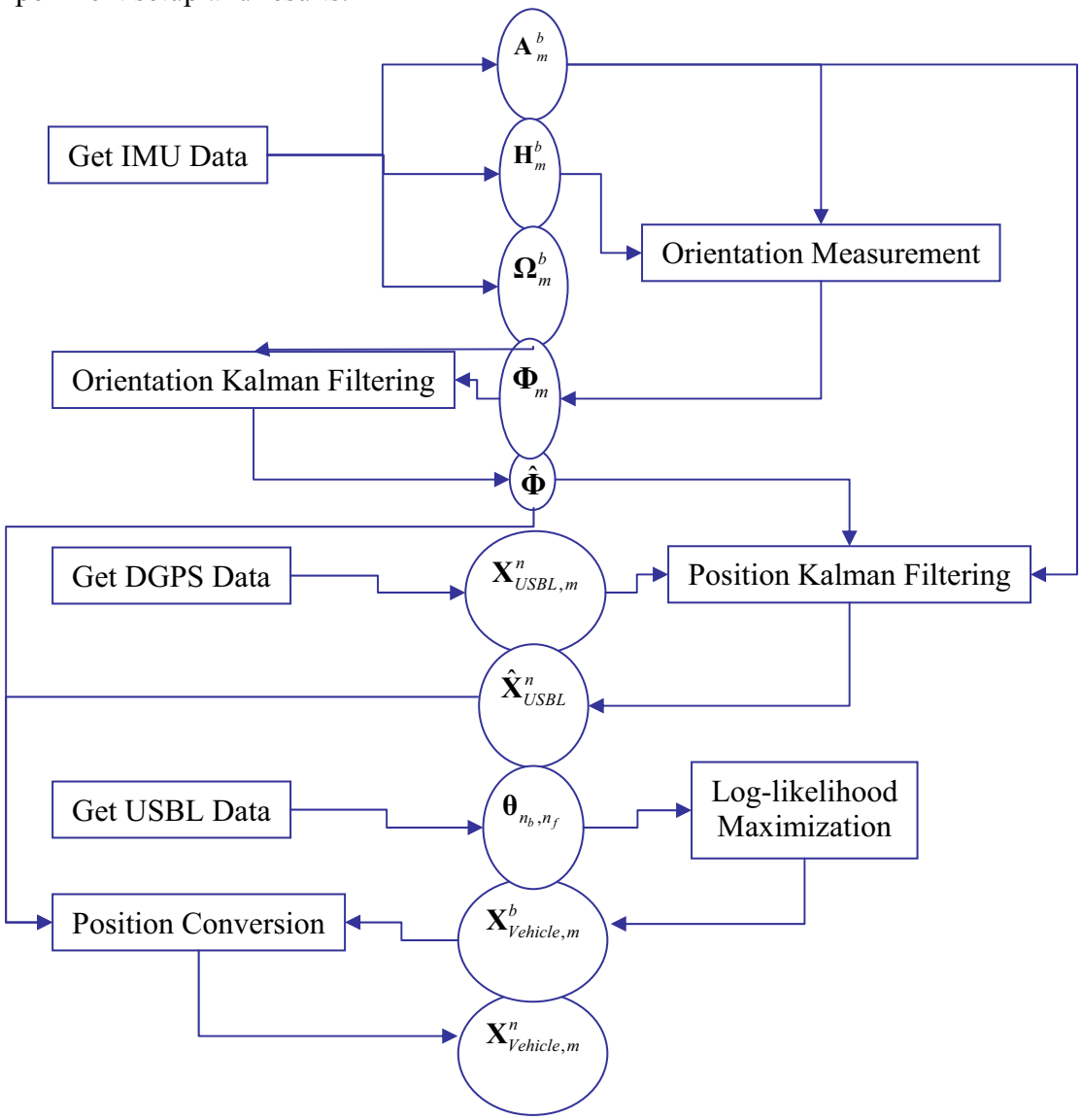


Figure 2.3.8. USBL-APS functional diagram [19].

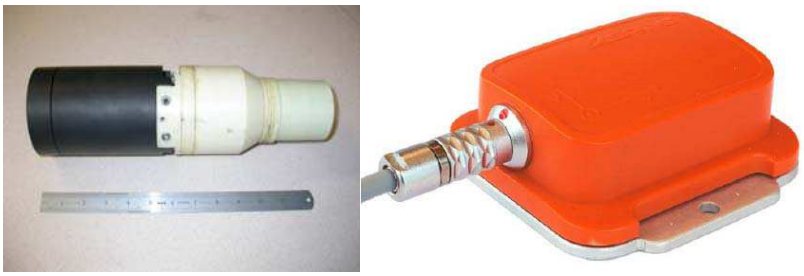


Figure 2.3.9. Coupled IMU and USBL Array (left) and XSens MTi IMU (right).

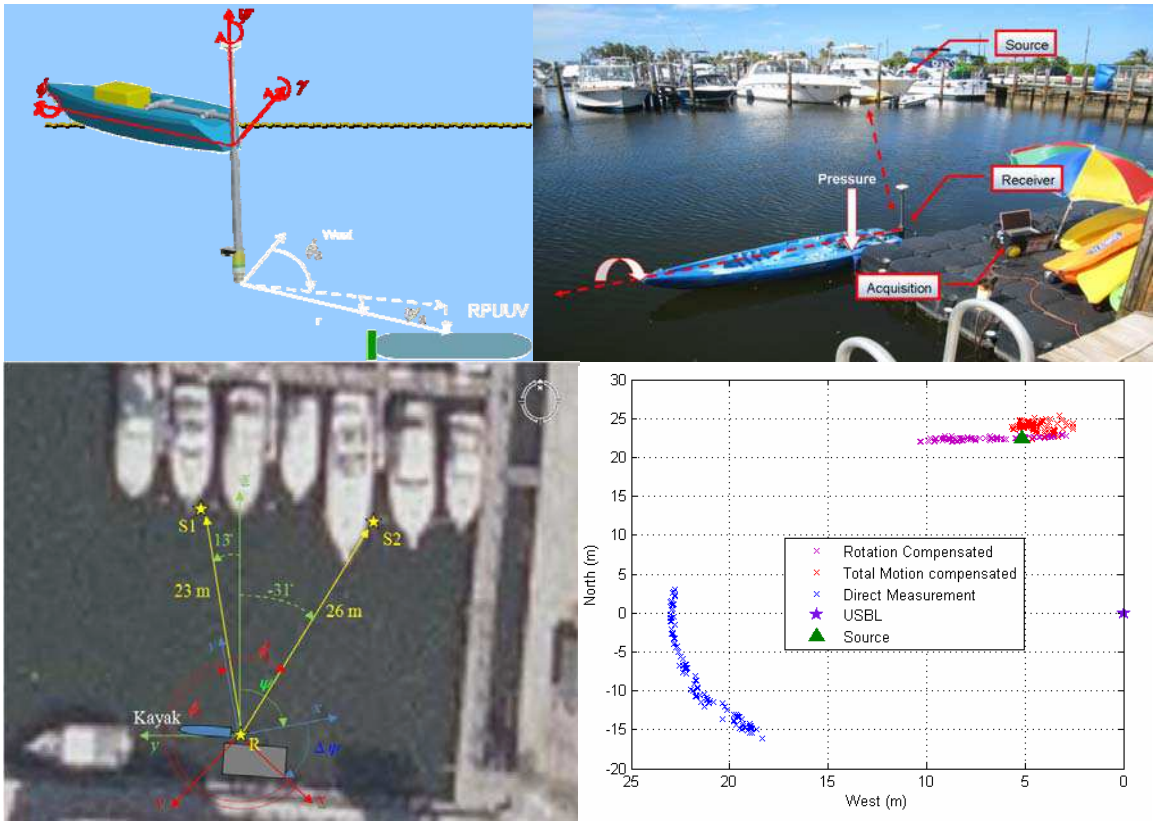


Figure 2.3.10. Acoustic positioning experiment at the FAU SeaTech Marina, Port Everglades, Florida: (top left) experimental setup, (top right) dock view of the experiment, (bottom left) aerial view of the experiment, (bottom right) source position estimation using the USBL-APS unit.

2.3.4 High-speed acoustic communications:

The high-speed high-frequency acoustic modem (HS-HFAM) technology presented in this section has been developed in part under separate funding from the Office of Naval Research, Science and Technology (code 32). The resulting financial and technical leverage allowed us to obtain the results shown here.

The one-way HS-HFAM, operating between 260 kHz and 380 kHz, has been developed to transmit compressed images from an underwater vehicle to a surface operator [11-16][18][21-22]. High data rates are made possible using a high-resolution decision

feedback equalizer with parallel algorithm for tracking and compensating large Doppler, developed at Florida Atlantic University (FAU). Two prototypes have been developed at FAU in partnership with EdgeTech, Inc. The source units are small (0.5 m in length by 0.12 m in diameter), lightweight and power-efficient. The receiver unit is a small, self-contained, lightweight and splash-proof case combined with a laptop. Small, single channel commercial transducers are used to transmit and receive the broad-band acoustic sequences. The HS-HFAM is a one-way high-speed acoustic modem designed to transmit combined images and text information in very short bursts. The HS-HFAM communication system accommodates multiple inputs and outputs using Ethernet connections. The inputs and outputs are either UDP or TCP-IP ports.

The three main concerns associated with broadband acoustic communication at high frequencies in harbors are reverberation, Doppler shift and, to a lesser extent, noise. Acoustic reverberation, originating from the scattering of acoustic waves off the surface, bottom, walls and obstacles, causes inter-symbol interference (ISI). In the frequency domain, reverberation is equivalent to frequency-selective fading. Frequency-selective fading also includes the effect of sound refraction due to sound velocity gradient. Doppler shift is due to the relative motion of the communication platforms and boundaries, especially the water surface ship hulls and some biological life. The combined effect of various Doppler shifts is known as Doppler spread. Background noise due to boat traffic is relatively benign around 300 kHz (approximately 35 dB re 1μPa/√Hz), however thermal noise causes an increase of 6 dB per octave above 100 kHz. The use of high-frequencies for high-speed underwater acoustic communications has significant advantages. First of all, the transducers are small, efficient and can be fitted in small UUVs. Also, the high bandwidth means high data rate and also excellent dual space-time resolution. With this high spatial resolution, DFE (Decision Feedback Equalizing) processes can better compensate the multipath, which is the main cause of limitation of this type of communication devices.

Each message contains three distinct parts used to detect, synchronize, identify and transfer encoded data, while ensuring efficient, error-free reception of the data. The first portion of a message is a 2.7 ms chirp transmitted between 247 and 273 kHz, with a dead-time of 3 ms, and used for detection and synchronization. The second portion is a 5.1 ms message header, which contains the symbol duration (40μs, 20μs, 13μs) and the type of modulation used (BPSK, QPSK). The data packet is received 3 ms after the message header. The number of information bits is set to 9120 plus 32 CRC bits, coded with BCH(15,11,1). The message starts with a 512-bit training sequence. The actual packet duration varies from 91.5 ms to 549.1 ms depending on the modulation. The true information bit rate varies from 16243 bps to 87768 bps. A tone is transmitted simultaneously at 375 kHz for efficient Doppler tracking. The HS-HFAM is remarkably power efficient: at full acoustic power and at the fastest bit rate, 13298.2 bits of information are transmitted per 1 Joule of acoustic energy. Table 2.3.1 summarizes the salient characteristics of the data packet.

HS-HFAM DATA PACKET SPECIFICATIONS

Modulation Type	BPSK	BPSK	BPSK	QPSK	QPSK	QPSK
Symbol Duration	40 μs	20 μs	13 μs	40 μs	20 μs	13 μs
Symbol Bandwidth	25 kHz	50 kHz	75 kHz	25 kHz	50 kHz	75 kHz
Information bits/ frame	1140	1140	1140	1140	1140	1140
Packet duration (ms)	0.5491	0.2745	0.1830	0.2745	0.1373	0.0915
Message duration (s)	0.5615	0.2869	0.1954	0.2869	0.1497	0.1039
Information rate (bps)	16243	31784	46668	31784	60935	87768
Packet coded rate (bps)	25000	50000	75000	50000	100000	150000
Bits-per-Joule (bit/J)	2461.1	4815.8	7070.9	4815.8	9232.6	13298.2

Table 2.3.1. High-speed high-frequency acoustic modem message specifications.

The data are collected using a high-resolution, low-noise acquisition system developed by EdgeTech Inc. in collaboration with FAU (Figure 2.3.11). The acquisition system produces complex base-band signals with a 24-bit resolution. These data are processed with a commercial off-the-shelf PC laptop, connected to the acquisition unit via the Ethernet. Each incoming message is detected, authenticated, equalized and decoded, and the output is relayed to a de-multiplexer which routes relevant information to each application. At present, the applications are the imaging sonar topside display and the vehicle control display.

Figure 2.3.11. HS-HFAM source (left) and receiver (right, courtesy of EdgeTech Inc.).

The HS-HFAM source was installed in the RPUV and transmitted snippets of canned information to the HS-HFAM receiver. These acoustic images were received at a rate of 4 per second during a set of experiments in the FAU SeaTech marina, in February 2008. The vehicle moved at a maximum speed of 0.5 m/s. The RPUV was piloted acoustically during this set of experiments, and no loss of performance was noticed either in terms of low-frequency acoustic piloting or in terms of high-frequency acoustic data transmission. The maximum distance from between the vehicle and the HS-HFAM receiver was 60 m. Figure 2.3.12 shows an aerial view of the experiment. Figure 2.3.13 shows the operator piloting the vehicle and the RPUV at various locations.

An example of compressed canned image is shown in Figure 2.3.14. The canned image and sensor information were initially collected with a Sound Metrics DIDSON sonar, and stored in the high-speed high-frequency acoustic modem memory. The image is displayed using the Sound Metrics DIDSON Viewer display.

The six modes of transmission listed in Table 2.3.1. For each transmission mode, the channel impulse response, Doppler shift due to the relative motion between source and receiver, minimum mean-square error (MMSE) at the equalizer output and bit error rate (BER) are displayed in Figures 2.3.15 to 2.13.20. Table 2.3.2 summarizes the BER measured vs. the transmission mode.

HS-HFAM PERFORMANCE

Modulation Type	BPSK	BPSK	BPSK	QPSK	QPSK	QPSK
Symbol Duration	40 µs	20 µs	13 µs	40 µs	20 µs	13 µs
Symbol Bandwidth	25 kHz	50 kHz	75 kHz	25 kHz	50 kHz	75 kHz
BER (%)	3.43%	2.07%	2.14%	6.38%	4.81%	4.14%

Table 2.3.2. Measured high-speed high-frequency acoustic modem performance vs. transmission mode.

Figure 2.3.12. Aerial view of the acoustic piloting and high-speed acoustic transmission using both the HS-HFAM and Acoustic Piloting Modem mounted on the RPUV.

Figure 2.3.13. Dock view of the acoustic piloting and high-speed acoustic transmission using the RPUV: (top left) operator piloting the vehicle, (top right) RPUV in the water at mission start, (bottom left) RPUV in the west channel, (bottom right) RPUV in the east dockage area.

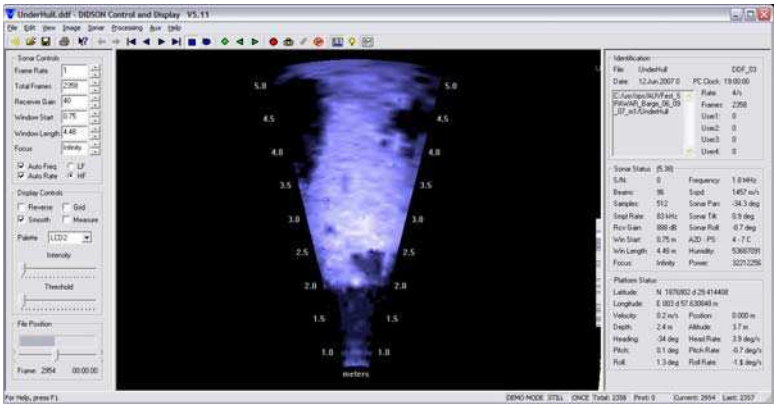


Figure 2.3.14. Compressed canned image and sensor information displayed on the Sound Metrics DIDSON viewer.

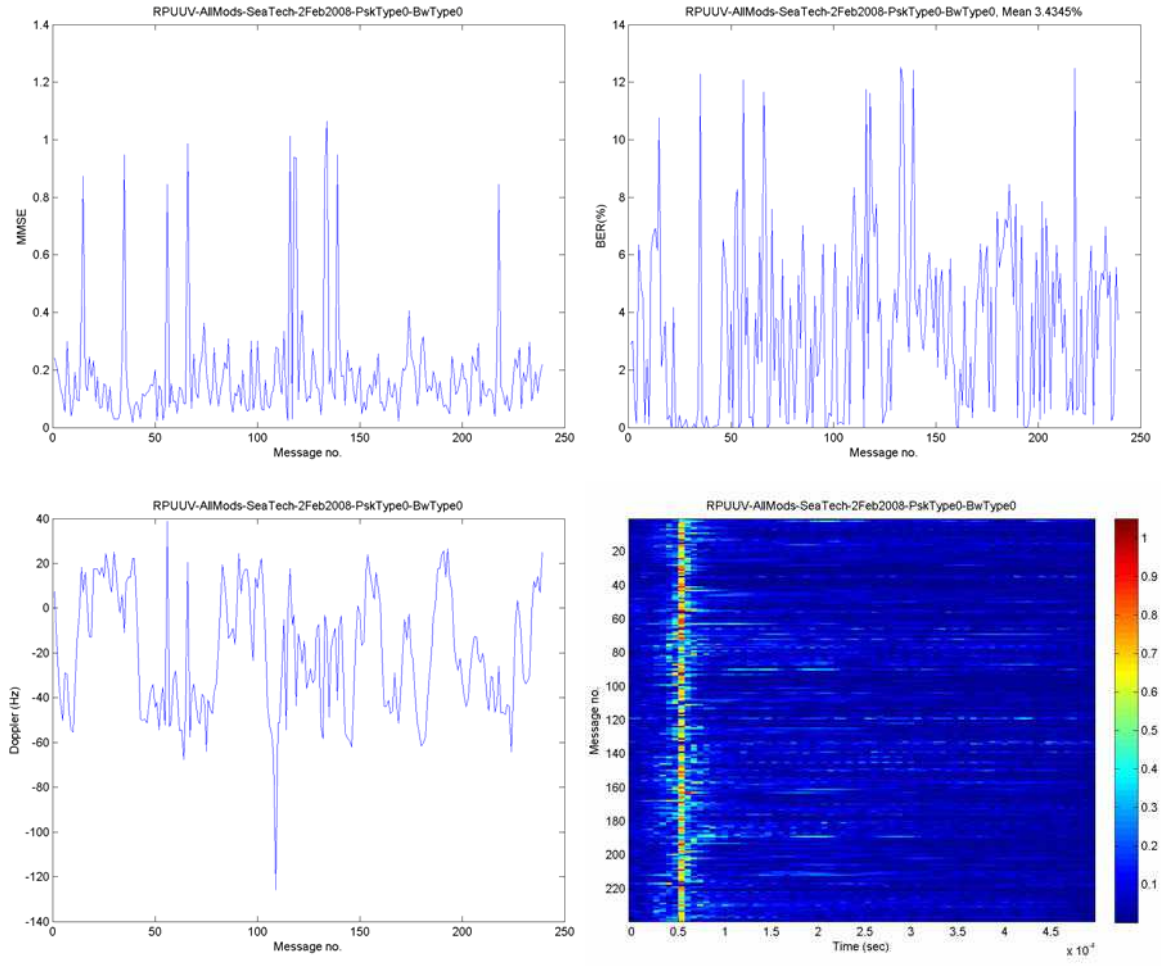


Figure 2.3.15. Experimental results using BPSK modulation and a symbol bandwidth of 25 kHz: (top right) minimum mean-square error (MMSE) at the equalizer output, (top left) bit error rate (BER), (bottom right) channel impulse response, (bottom left) Doppler shift due to the relative motion between source and receiver.

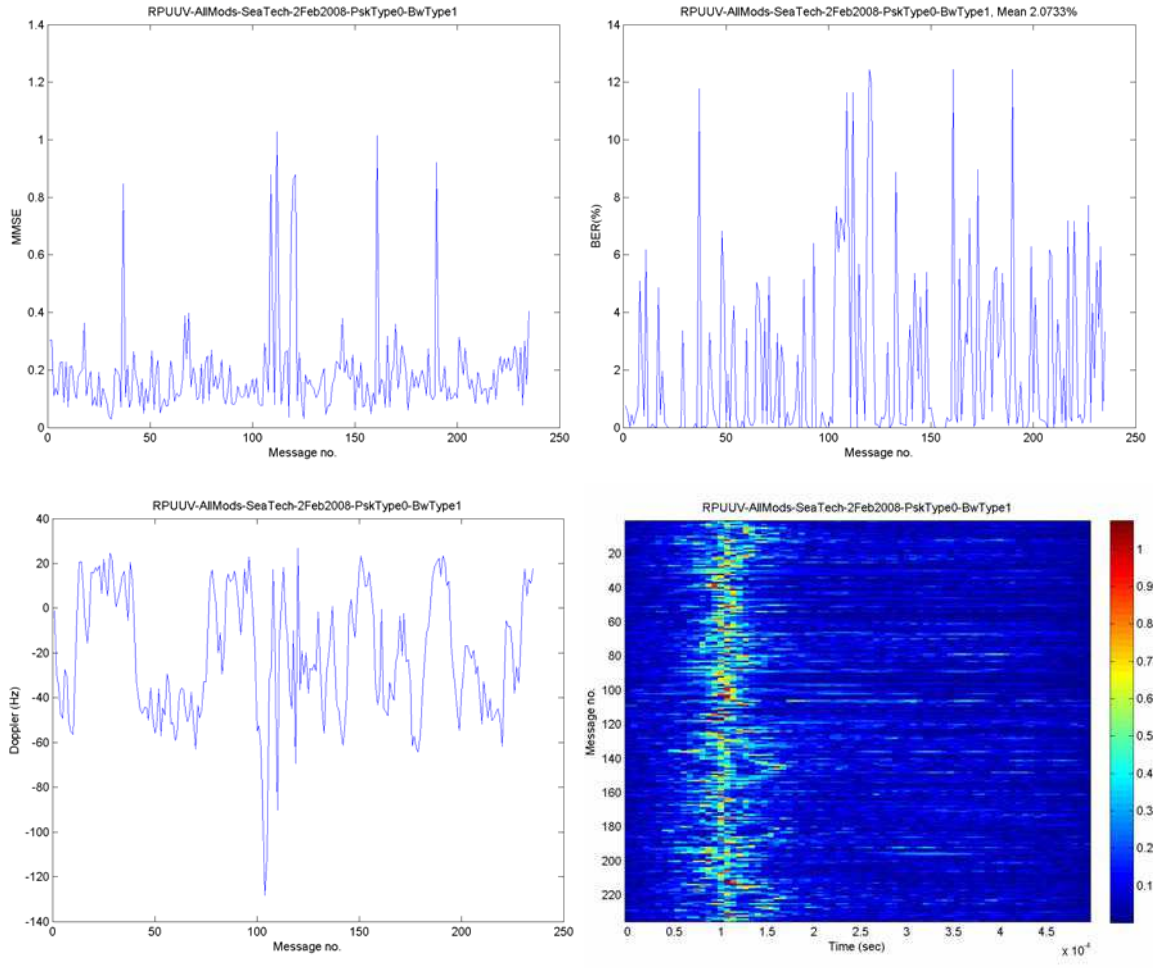


Figure 2.3.16. Experimental results using BPSK modulation and a symbol bandwidth of 50 kHz: (top right) minimum mean-square error (MMSE) at the equalizer output, (top left) bit error rate (BER), (bottom right) channel impulse response, (bottom left) Doppler shift due to the relative motion between source and receiver.

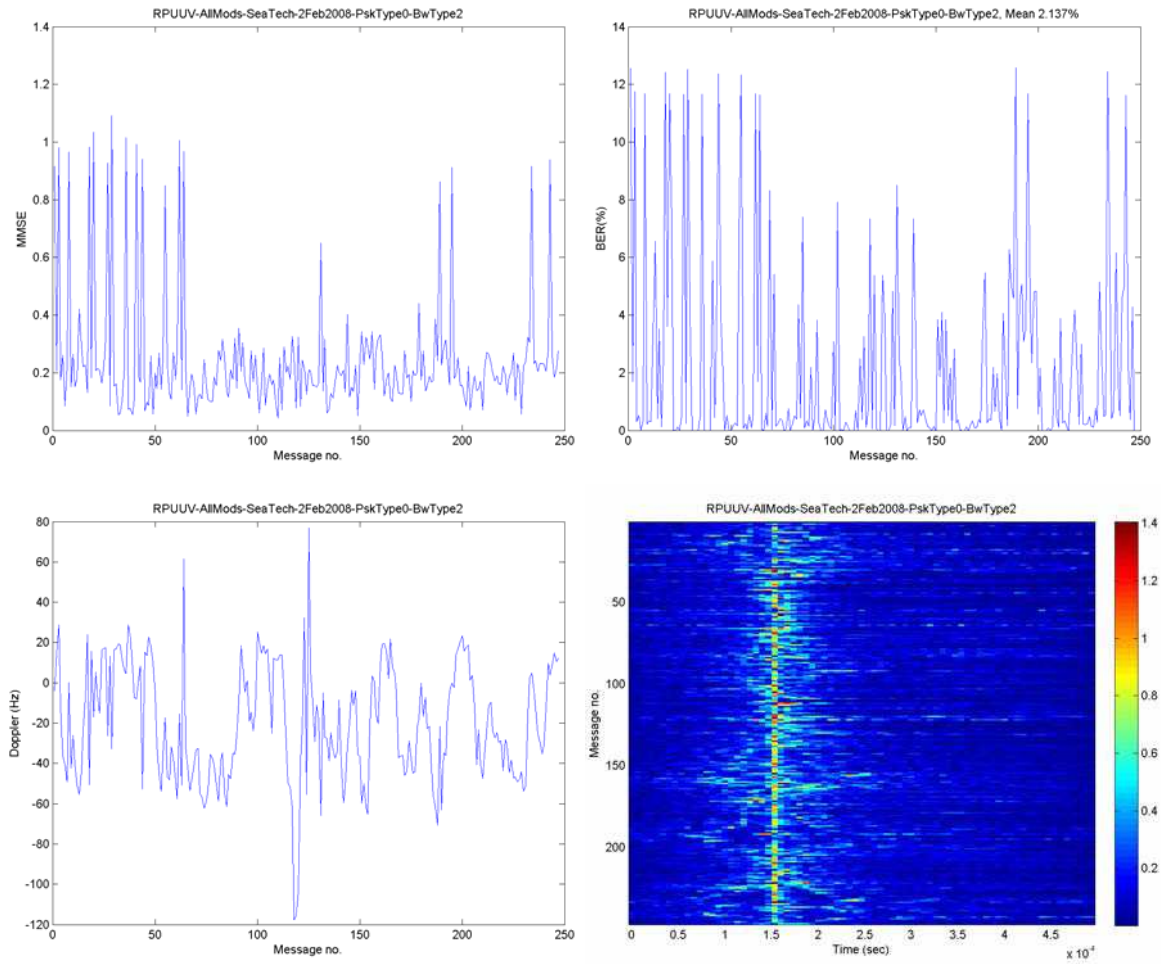


Figure 2.3.17. Experimental results using BPSK modulation and a symbol bandwidth of 75 kHz: (top right) minimum mean-square error (MMSE) at the equalizer output, (top left) bit error rate (BER), (bottom right) channel impulse response, (bottom left) Doppler shift due to the relative motion between source and receiver.

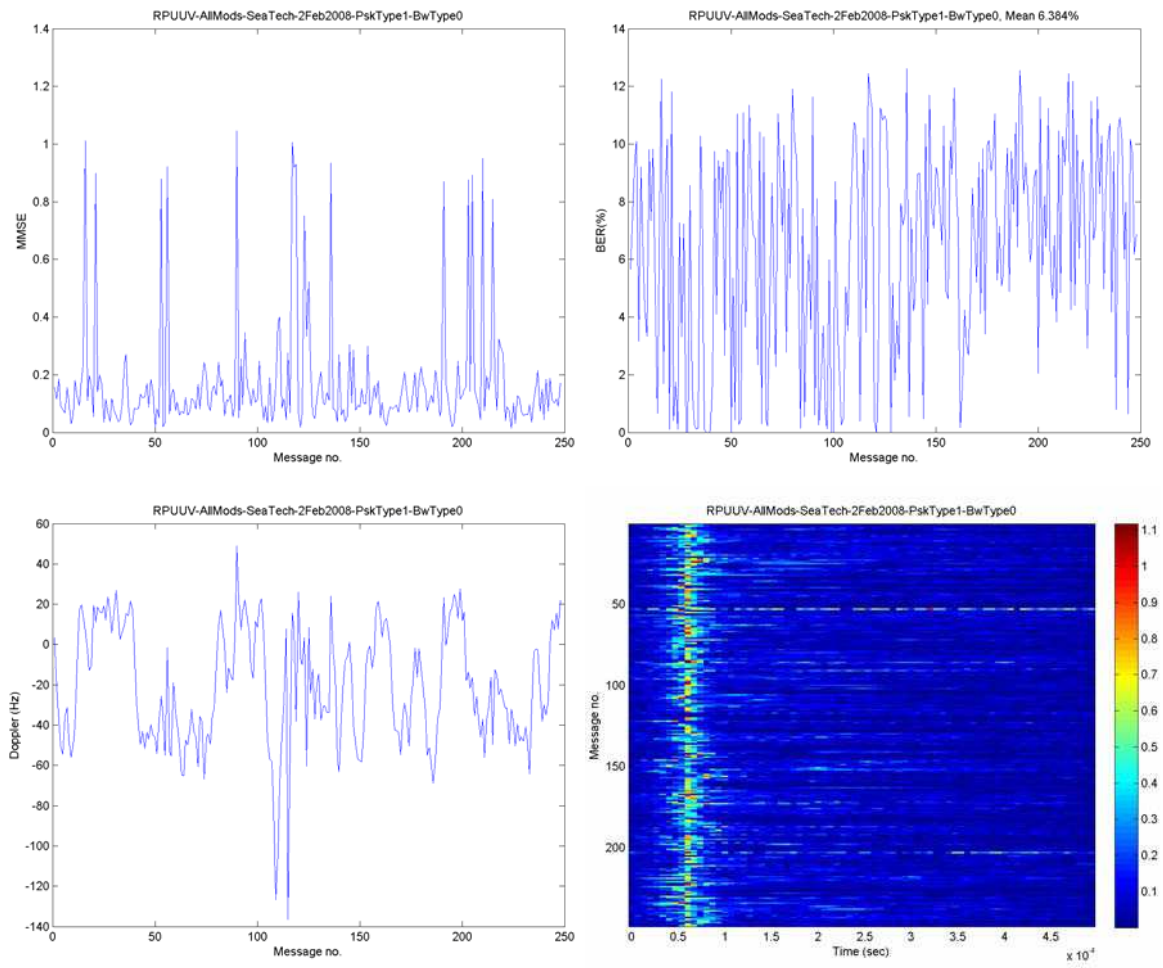


Figure 2.3.18. Experimental results using QPSK modulation and a symbol bandwidth of 25 kHz: (top right) minimum mean-square error (MMSE) at the equalizer output, (top left) bit error rate (BER), (bottom right) channel impulse response, (bottom left) Doppler shift due to the relative motion between source and receiver.

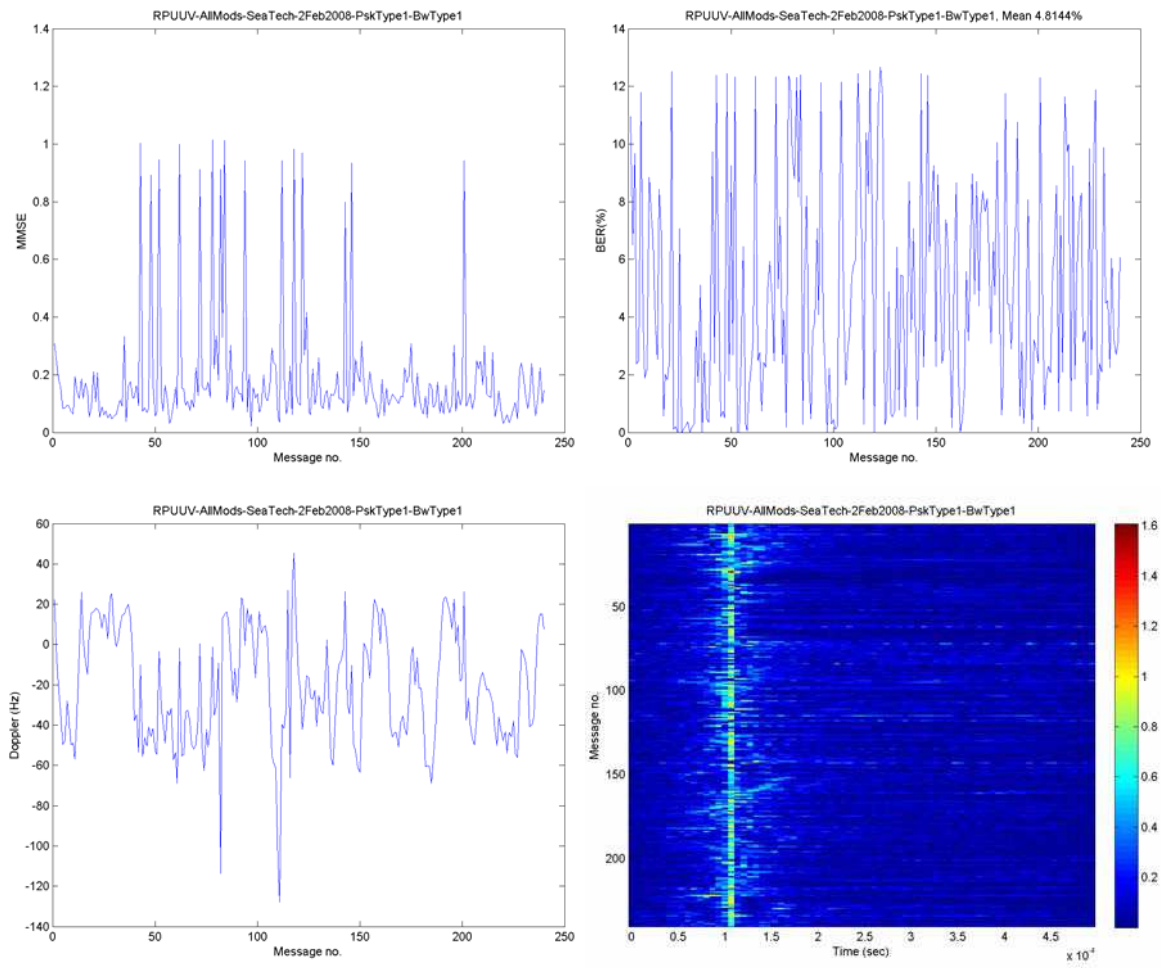


Figure 2.3.19. Experimental results using QPSK modulation and a symbol bandwidth of 50 kHz: (top right) minimum mean-square error (MMSE) at the equalizer output, (top left) bit error rate (BER), (bottom right) channel impulse response, (bottom left) Doppler shift due to the relative motion between source and receiver.

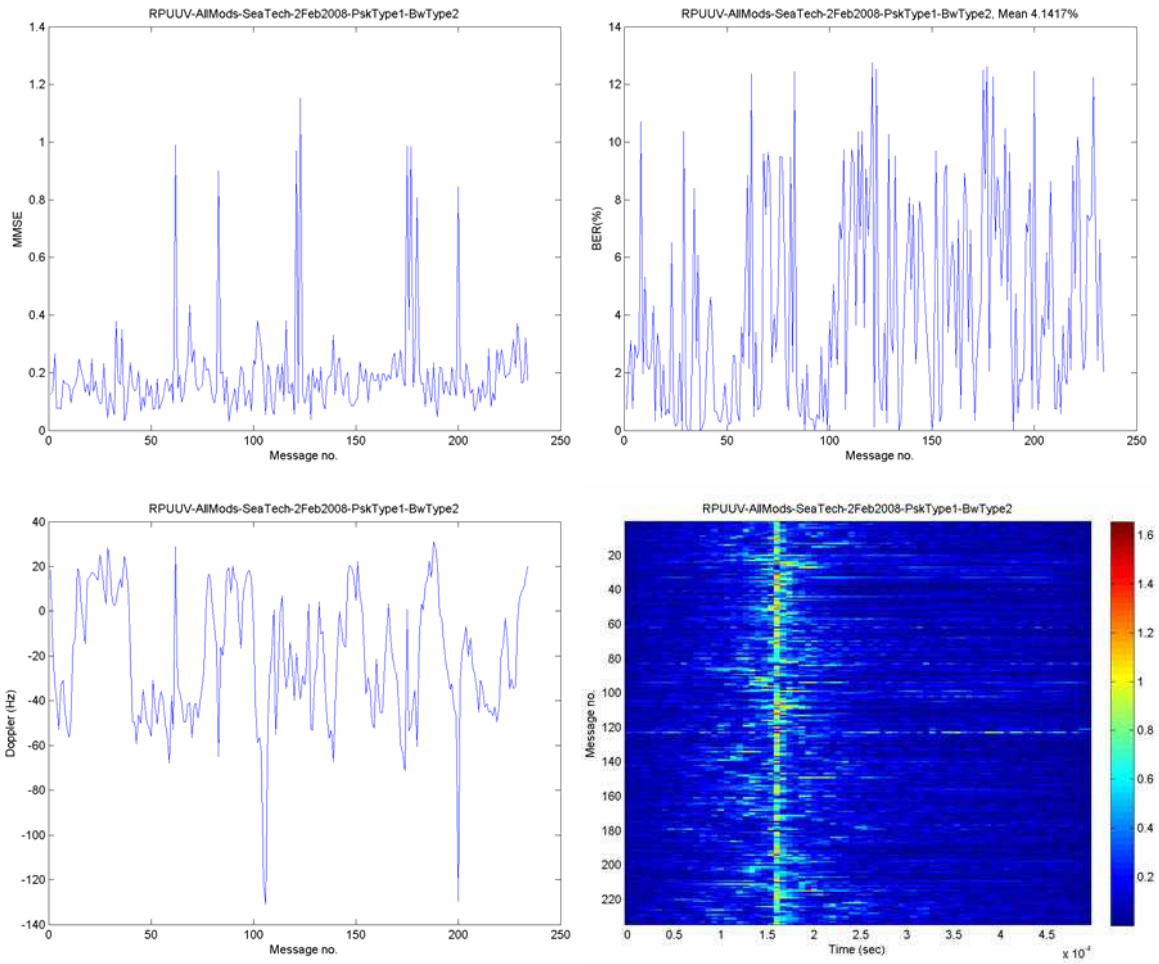


Figure 2.3.20. Experimental results using QPSK modulation and a symbol bandwidth of 75 kHz: (top right) minimum mean-square error (MMSE) at the equalizer output, (top left) bit error rate (BER), (bottom right) channel impulse response, (bottom left) Doppler shift due to the relative motion between source and receiver.

References

[1] Center for Coastline Security Technology, "Year One: Final Technical Report," Florida Atlantic University and US ONR, June, 2006.
[2] R. Uhrich and J. Walton, "Supervisory Control of Untethered Undersea Systems: A New Paradigm Verified," presented at The 9th International Symposium on Unmanned Untethered Submersible Technology, Sept. 1995.
[3] H. Robinson and A. Keary," Remote Control of Unmanned Undersea Vehicles," presented at the International Unmanned Undersea Vehicle Symposium, April 2000.
[4] M. Dunbabin, J. Roberts, K. Usher, G. Winstanley and P. Corke, "A Hybrid AUV Design for Shallow Water Reef Navigation," presented at the International Conference on Robotics and Automation, April 2005.
[5] A. M. Polsenberg, "Developing an AUV Manual Remote Control System," M. S. Thesis, Massachusetts Institute of Technology, 2000.

[6] P. P. J. Beaujean, "Short Course on Underwater Acoustic Communications", 151st Meeting of the Acoustical Society of America, 5-9 June 06, Providence, Rhode Island.
[7] P.P.J. Beaujean and E.P. Bernault, "A New Multi-Channel Spatial Diversity Technique for Long Range Acoustic Communications in Shallow Water", MTS/IEEE Oceans'03 Proc., San Diego, CA, Sept. 2003.
[8] P.P.J. Beaujean, A.I. Mohamed and R. Warin, **"**Maximum Likelihood Estimates of a Spread-Spectrum Source Position using a Tetrahedral Ultra-Short Baseline Array", Proceedings of OES/IEEE Oceans'05 Europe, June 2005, Brest, France.
[9] P.P.J. Beaujean, A.I. Mohamed and R. Warin, "Acoustic Positioning using a Tetrahedral Ultra-Short Baseline Array of an Acoustic Modem Source Transmitting Frequency-Hopped Sequences", Journal of Acoustical Society of America, J. Acoust. Soc. Am. 121, 144 (2007).
[10] R. G. Brown, P. Y.C Hwang,"Introduction to Random Signals and Applied Kalman Filtering", Third Edition", John Wiley & Sons.
[11] P.P. Beaujean, T. Nguyen, "High-Speed, High-Frequency Acoustic Modem (HS-HFAM) coupled with a Wavelet-Based Embedded Compression", ONR Joint Review of Unmanned Systems Technology Development, Panama City, Florida, Feb. 2006.
[12] P.P. Beaujean, "High Data Rate, Short Range Communications", ONR Joint Review of Unmanned Systems Technology Development, Panama City, Florida, Feb. 2005.
[13] J. Spruance, P.P. Beaujean, "A High-Speed High-Frequency Broadband Acoustic Modem for Short-to-Medium Range Data Transmission in Ports, Very Shallow Waters and Deep Waters using Spread-Spectrum Modulation and Decision Feedback Equalizing", Phase-I STTR, final report, March 2006.
[14] J. Spruance, P.P. Beaujean, "A High-Speed High-Frequency Broadband Acoustic Modem for Short-to-Medium Range Data Transmission in Ports, Very Shallow Waters and Deep Waters using Spread-Spectrum Modulation and Decision Feedback Equalizing", Phase-I STTR, 2nd status report, December 2005.
[15] J. Spruance, P.P. Beaujean, "A High-Speed High-Frequency Broadband Acoustic Modem for Short-to-Medium Range Data Transmission in Ports, Very Shallow Waters and Deep Waters using Spread-Spectrum Modulation and Decision Feedback Equalizing", Phase-I STTR, 1st status report, September 2005.
[16] P.P.J. Beaujean, P.M. Blue and D. Kriel, "A High-Speed High-Frequency Acoustic Modem (HS-HFAM) for Ports and Shallow Water Operation", 13th International Congress on Sound and Vibrations, Vienna, Austria, July 2006.
[17] Center for Coastline Security Technology, "Year Two: Final Technical Report," Florida Atlantic University and US ONR, June, 2007.
[18] Patrick Blue, "High-Speed, High-Frequency Acoustic Modem for Short-Range Underwater Communications in Ports and Shallow Waters", Master's thesis student in Ocean Engineering, sponsored by the Center for Coastline Security and Technology.
[19] Antoine Bon, "Motion-Compensated Acoustic Positioning of an Acoustically Piloted Underwater Vehicle Using an Ultra Short Baseline Mounted on a Moving Platform", Master's thesis student in Ocean Engineering, sponsored by the Center for Coastline Security and Technology.
[20] Matthew Staska, "A Study of the Underwater Acoustic Propogation in a Turning Basin Modeled as a Three Dimensional Duct Closed at One End using the Method Of

Images", Master's thesis student in Ocean Engineering, sponsored by the Center for Coastline Security and Technology and recipient of the FAU College of Engineering Dean's Award.
[21] P.P.J. Beaujean, "Real-Time Image and Status Transmission from a UUV during a Ship Hull Inspection in a Port Environment using a High-Speed High-Frequency Acoustic Modem", *Proceedings of MTS/IEEE Oceans'07*, October 2007, Vancouver, CA.
[22] P.P.J. Beaujean, "High-Speed High-Frequency Acoustic Modem for Image Transmission in Very Shallow Waters", *Proceedings of OES/IEEE Oceans'07 Europe*, June 2007, Aberdeen, UK.

2.4 Environmental Assessment and Modeling: Monitoring Currents and Ambient Noise in Ports and Data Synthesis

PI: Dr. George V. Frisk

Tasks 3.9-3.12

2.4.1 Summary

A methodology for characterizing the acoustical properties of a port environment, namely Port Everglades, has been proposed and carried out. This approach includes both a port-wide analysis of how the basic oceanographic features within the port impact the acoustic properties, and also a more focused sampling methodology within a small region of Port Everglades, allowing for the acoustic characteristics, including ambient noise, and an approximate signal absorption to be computed. The results documented through the duration of this research indicate that the temperature variation throughout the port is the principal contributor to the characteristics of the sound velocity profile. Ambient noise measurements have revealed high levels of background noise within the sub-5 kHz region, owing likely to consistent port traffic. The calculation of absorption indicates that high frequency systems, i.e. >100 kHz, may encounter problems when transmitting over a considerable distance. These are important factors for consideration when implementing a successful underwater acoustic system.

The development of an unmanned underwater vehicle at Florida Atlantic University with onboard optical sensors has prompted the temporal and spatial optical characterization of Port Everglades, with in-situ measurements of the turbidity, conductivity, and temperature. Water samples were collected for laboratory analysis where attenuation and absorption were measured with a bench top spectrometer. All of the measurements showed a high degree of variability within the port on a temporal and spatial basis. Correlations were researched between the measured properties as well as tide and current. Temporal variations showed a high correlation to tidal height but no relation was found between turbidity and current, or salinity. As a result, the planned current measurement program was not conducted. Spatial variations were primarily determined by proximity to the port inlet. Proportionality constants were discovered to relate turbidity to scattering and absorption coefficients, as well as visibility. These constants along with future turbidity measurements will allow the optimization of any underwater camera system working within these waters.

2.4.2 Introduction

The overall goal of this project is to characterize the acoustical, oceanographic, and optical properties of Port Everglades as they relate to the operation of the Remotely Piloted Unmanned Underwater Vehicle (RPUUV) being developed at the Center for Coastline Security Technology (CCST). Once the relation of these properties to the

functionality of the RPUUV sonar and video systems is adequately understood, this approach can be applied to other port environments in which similar surveillance systems may be employed.

2.4.3 Acoustical and Oceanographic Characteristics of Port Everglades

The results obtained from carrying out a series of environmental characterization experiments in Port Everglades are to be detailed and explained here. The first section of the results is obtained from assessing Port Everglades on a port-wide basis, and drawing conclusions on which oceanographic parameters impact the sound velocity characteristics, and how those characteristics vary both spatially and temporally throughout the port.

The second section will focus on the South Turning Notch of Port Everglades, which as explained in the previous chapter, is the location that has been chosen as the testbed for the FAU RPUUV that is currently being designed as part of the CCST project. The results from this region will illustrate the sound velocity characteristics obtained from completing a significantly more condensed profiling strategy. This, when combined with the ambient noise measurements recorded over a twelve hour period within the same region, will provide a thorough characterization of the South Turning Notch within Port Everglades.

The final section will describe the absorption calculations. As mentioned previously, within a turbid environment such as Port Everglades, an acoustic signal will not only be attenuated due to the water through which it travels, but also energy will be lost due to the absorption by suspended particles, more specifically, from viscous absorption and from scattering. Using the data obtained from the previous sections, combined with experimentally derived coefficients for calculating viscosity and density, an approximation for the total absorption of an acoustic signal is computed.

2.4.3.1 Port Everglades Sampling Strategy

In order to gain an understanding of how the basic oceanographic features such as temperature, pressure, and salinity vary as a function of space and time, and hence gain an understanding of the variation in sound velocity, approximately 200 different locations (cf. Fig. 2.4.1) have been profiled within Port Everglades using a Falmouth Scientific CTD instrument. This will allow for the variation in sound velocity as a function of time, position, and depth throughout the port to be investigated.

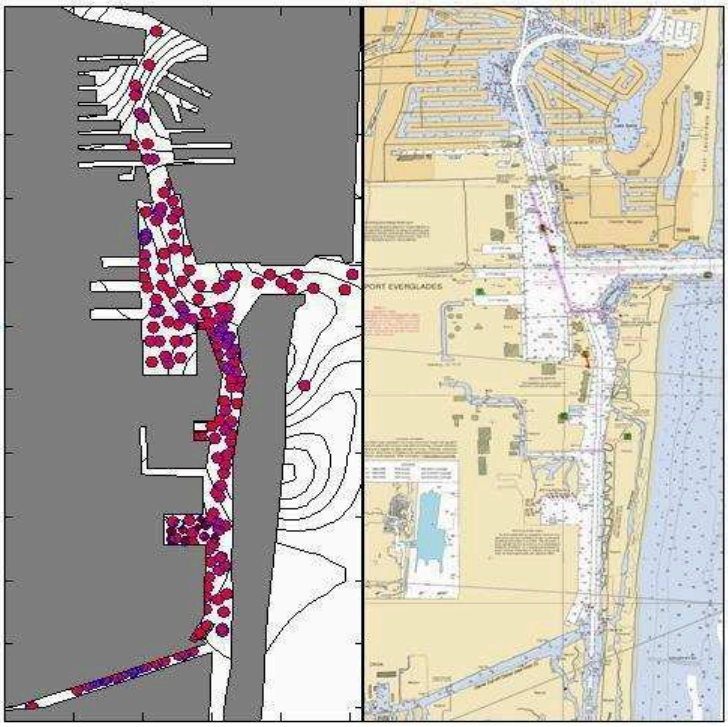


Figure 2.4.1 All Locations Profiled Within Port Everglades

2.4.3.2 Spatial and Temporal Variation

The following figures depict how the temperature, salinity and sound velocity vary throughout the Port at various depths. For illustrative purposes, this analysis focuses specifically on the profiles taken on 8th May 2006 to show the dependence of the sound velocity profile on the temperature and salinity components. Thus, this section focuses on the spatial variation throughout the Port on one day. The following section will investigate how the parameters, more specifically the sound velocity, vary temporally over the duration of the sampling period.

Figure 2.4.2 illustrates the temperature, salinity, and sound velocity variation on a Port-wide basis for 05-08-06 at a depth of 3 meters. The MATLAB program used to compute these plots illustrates the locations profiled by way of red circles, and interpolates the data between those points. As such, the values shown *outside* the port region are an inaccurate representation of the corresponding parameter within that location.

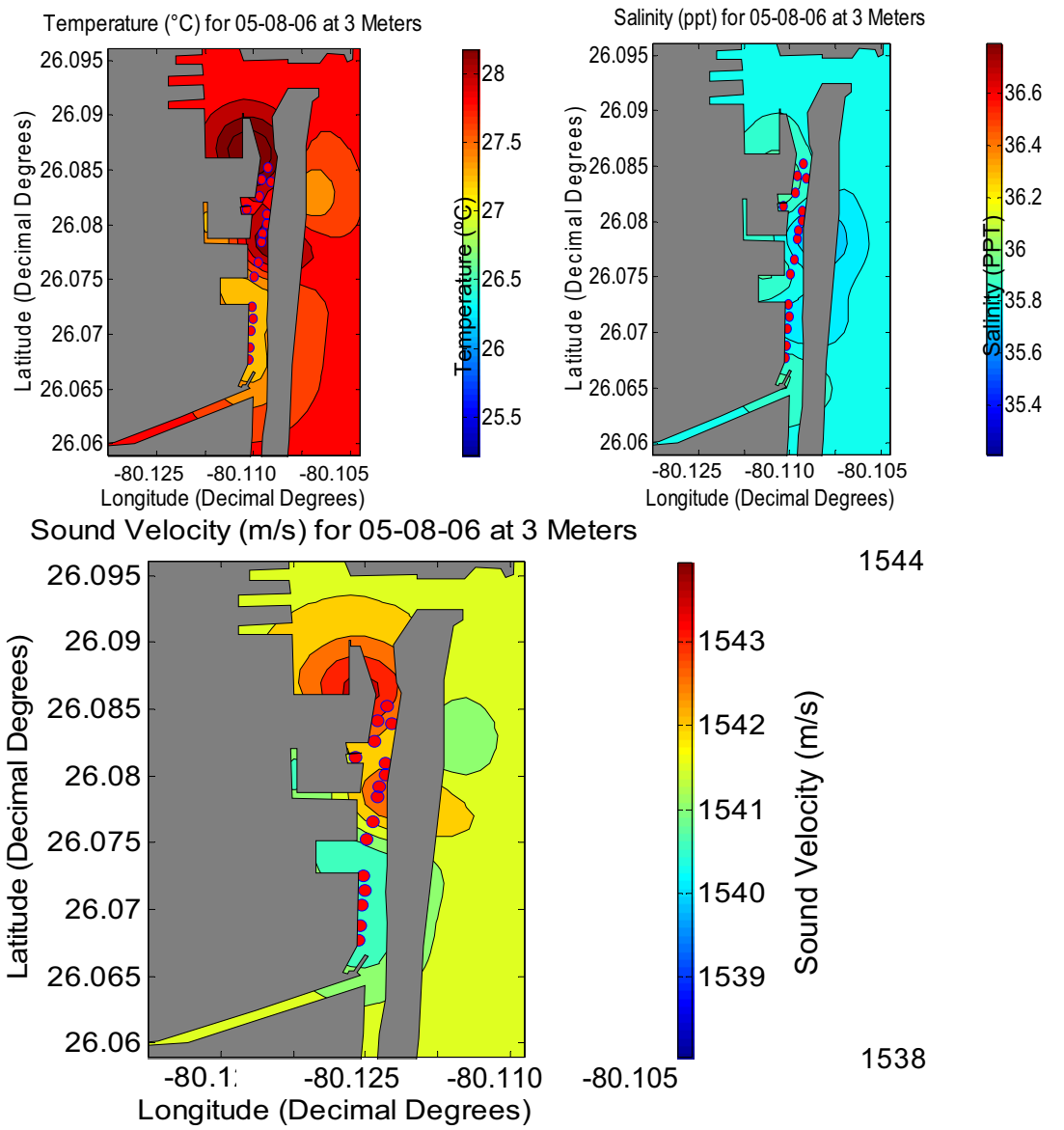


Figure 2.4.2 Spatial Variation of Sound Velocity, Temperature, and Salinity for 05-08-06 at Depth 3m

An examination of Fig. 2.4.2 shows that the contour shapes and the variation in sound velocity is closely matched to the variation displayed in the temperature plots.

Figure 2.4.3 illustrates the temperature, salinity, and the sound velocity variation on a port-wide basis for 05-08-06 at a depth of 13 meters. At this depth, the temperature and sound velocity profiles are also closely related.

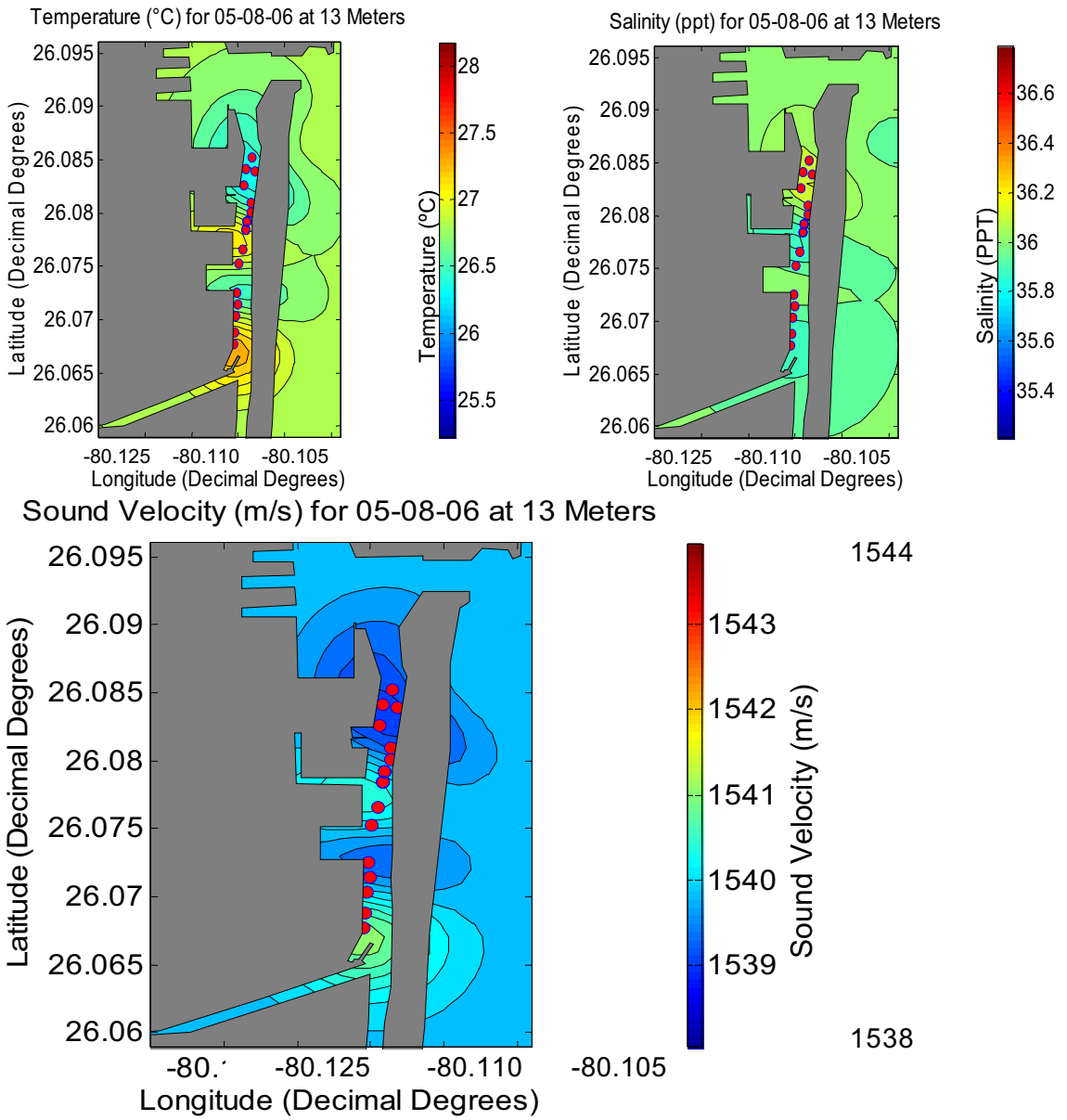


Figure 2.4.3 Spatial Variation of Sound Velocity, Temperature, and Salinity for 05-08-06 at Depth 13m

The port is a shallow region, and as a result it is advantageous to review the tides and currents during the period of sampling. The tide and current data for 08th May 2006 can be seen in Figs. 2.4.4 and 2.4.5, which are data taken from Captain Voyager Charting Software [3].

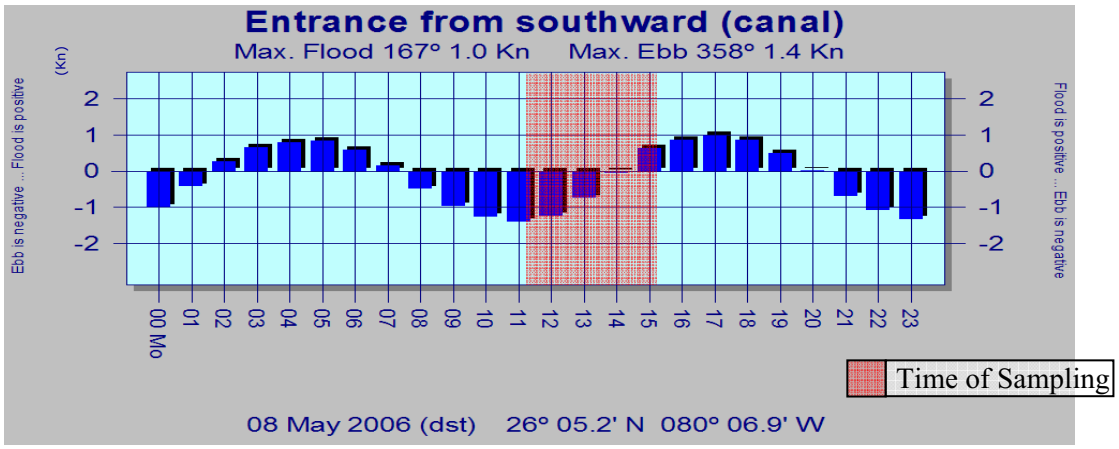


Figure 2.4.4 Current Chart For 05-08-06

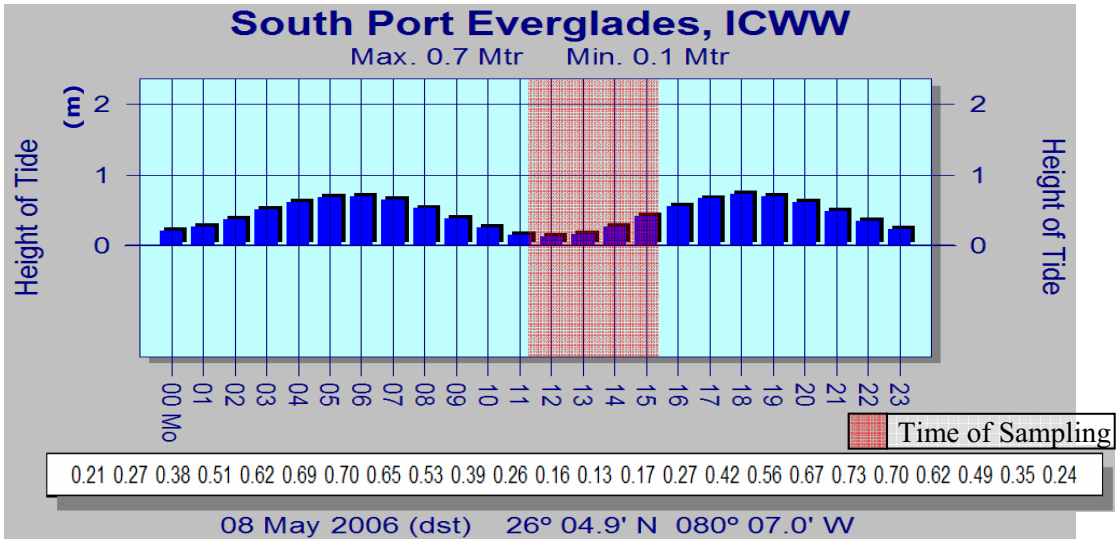


Figure 2.4.5 Tide Chart For 05-08-06

The change in tide and currents during the period of sampling for 08-05-06 is not significant, and as such the impact that these variations has on the varying profiles is minimal. A similar type of analysis was conducted for each day that was sampled through 2006 [1]. A review of these results indicates a trend comparable to that observed in the above profiles, and draws no correlation between the profiles recorded and the tide and current data.

After completing the analysis of the temperature, salinity and sound velocity profiles for 08th May 2006, along with the tide and current data, it is apparent that the sound velocity, in such shallow waters, is mostly driven by the temperature profile. A closer indication of this can be observed in Figs. 2.4.6 and 2.4.7. These plots clearly show the similarities between the sound velocity and temperature profiles. A detailed linear regression analysis confirms the strong correlation between sound velocity and temperature [1].

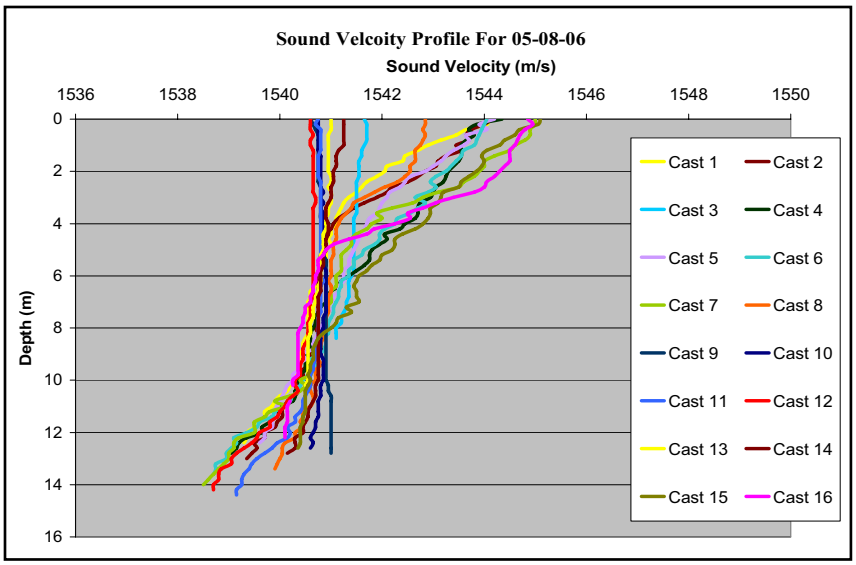


Figure 2.4.6 Sound Velocity Profile for All Casts Taken On 05-08-06

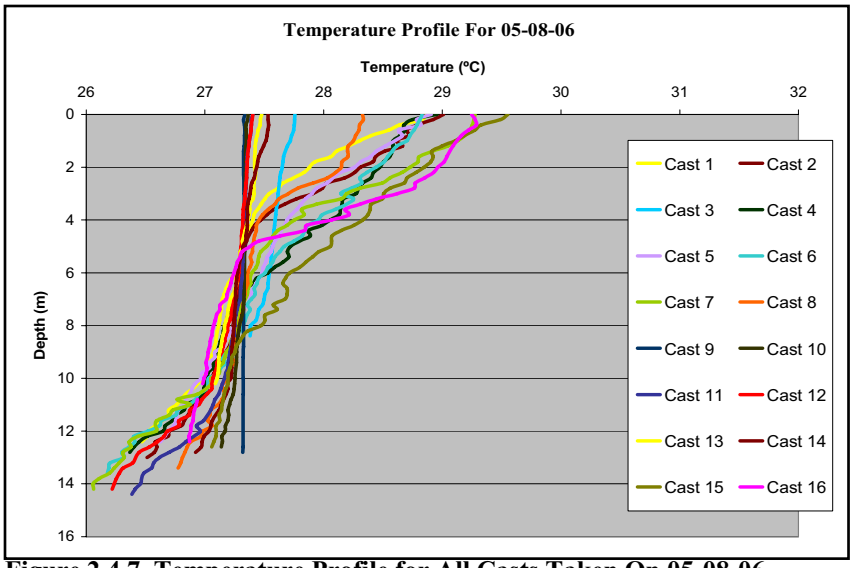


Figure 2.4.7 Temperature Profile for All Casts Taken On 05-08-06

Finally, it was also observed that the sound velocities increased with the time of sampling from March through May [1]. It is probable that the trend of increasing sound velocity with the time of year is to be attributed to the increase in water temperature due to the seasonal change.

2.4.3.3 Variability in Port Everglades South Turning Notch

As previously described, the RPUUV being developed by Florida Atlantic University is to be tested within the South Turning Notch of Port Everglades. In order to better understand the environment in which the vehicle is to be tested, a more condensed sampling strategy focused solely on the turning notch was completed, combined with a

twelve hour ambient noise sampling period. The specific locations that were profiled within the notch are shown in Fig. 2.4.8.

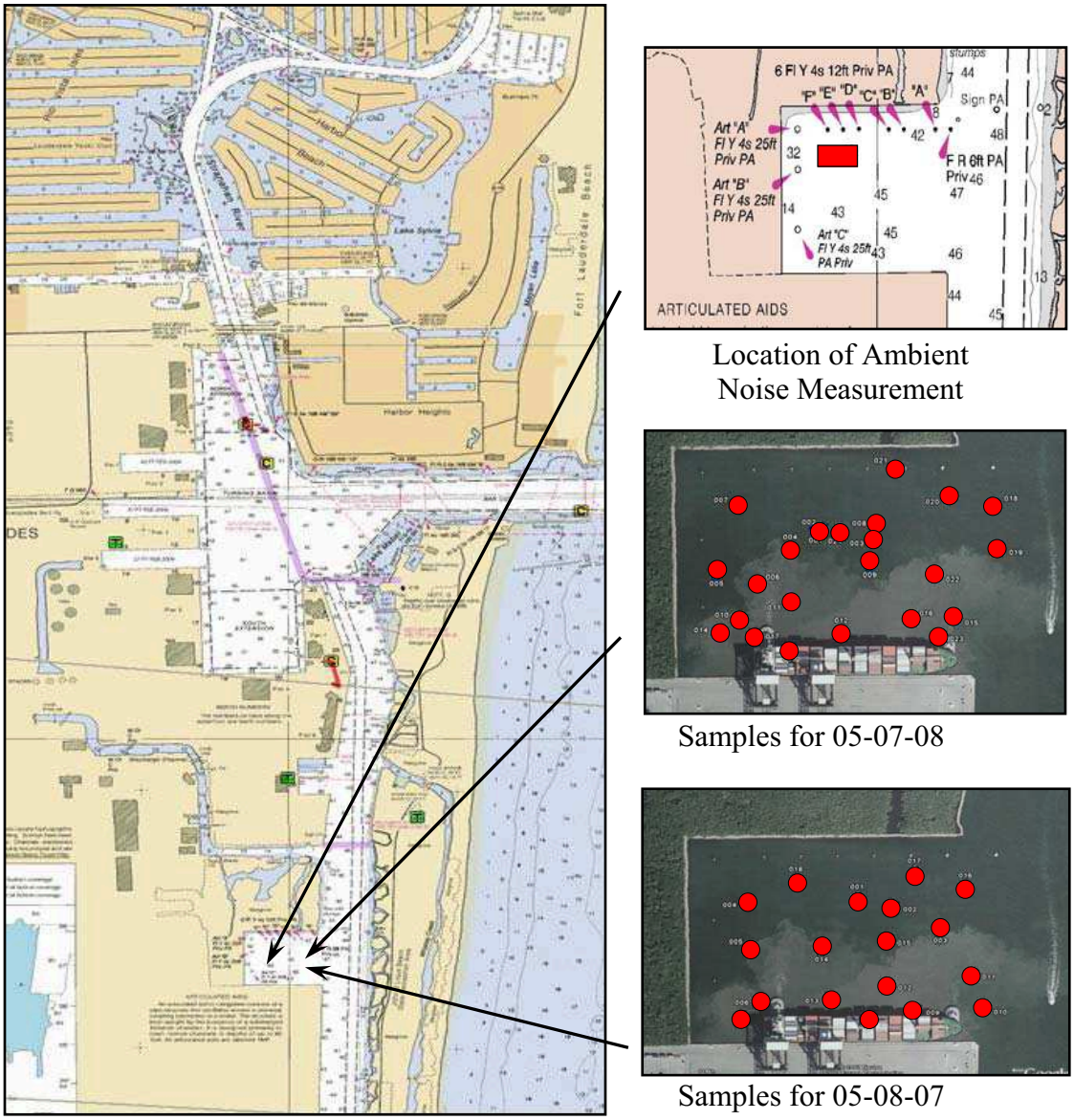


Figure 2.4.8 Specific Locations Profiled Within Port Everglades South Turning Notch

The data recorded from the concentrated sampling strategy can be observed in Figs. 2.4.9-2.4.12, where the sound velocity and temperature profiles from each cast and from each day are illustrated.

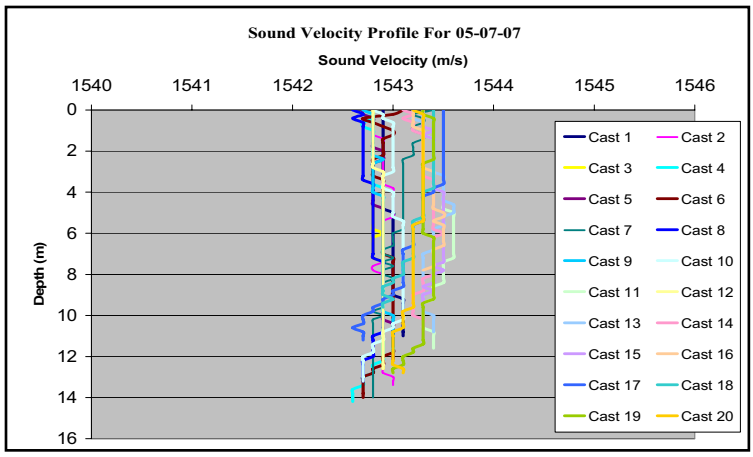


Figure 2.4.9 Sound Velocity Profiles for 05-07-07

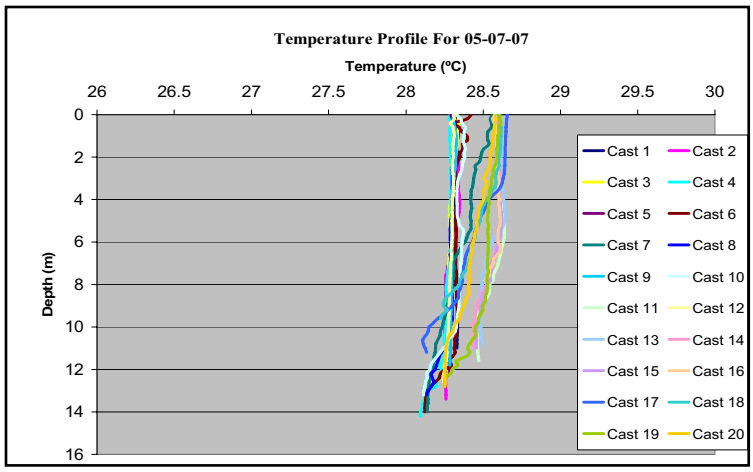


Figure 2.4.10 Temperature Profiles for 05-07-07

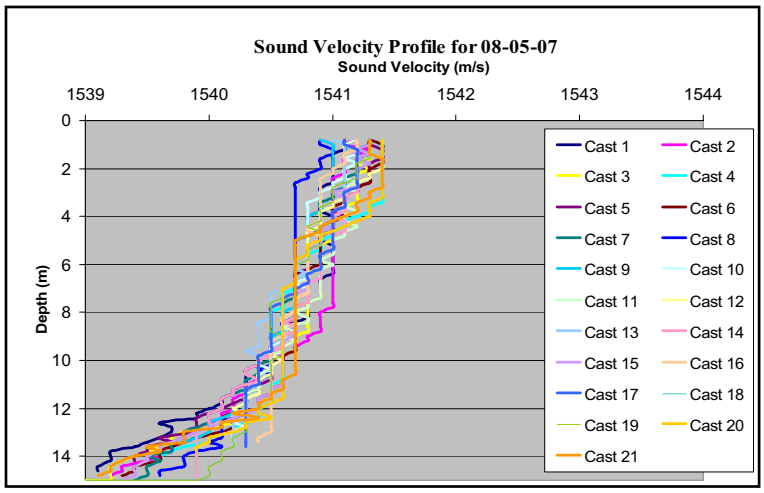


Figure 2.4.11 Sound Velocity Profiles for 05-08-07

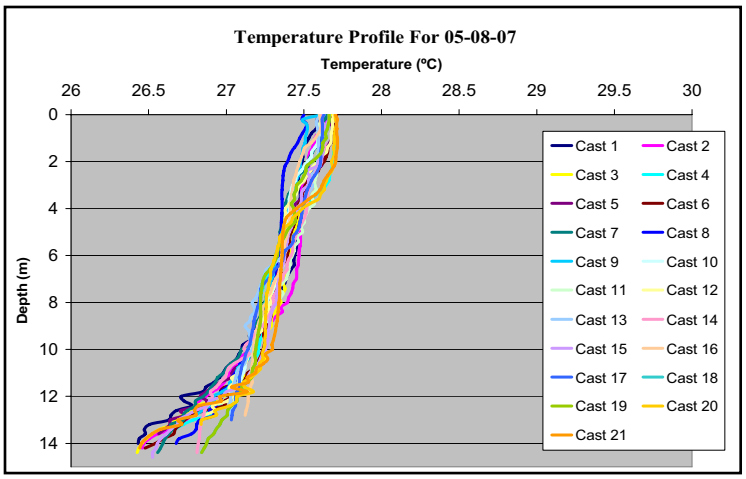


Figure 2.4.12 Temperature Profiles for 05-08-07

The sound velocity profiles generated from 07th May 2007 are clearly iso-velocity, varying a small amount along the water column. It should be noted that during this day of sampling the CTD casts were conducted following the exit of a large commercial vessel employing the use of two port tug vessels. This caused an extensive amount of mixing, resulting in the profiles observed above. The sound velocity characteristics recorded from 08th May 2007 are a more accurate representation of how the sound velocity varies with depth within this region. It can be seen that there is a slight decrease in sound velocity with depth to approximately 10 m, after which a stronger drop off is observed down to 15 m. The results obtained during this sampling period are similar to that recorded from the Port-wide assessment in that the temperature profile appears to be the main contributor to the sound velocity characteristics.

2.4.3.4 Ambient Noise Measurements

In order to analyze the ambient noise data recorded in the turning notch on 26th March 2007 in an efficient manner, 23 sample files were generated, each of 1.5 seconds in length, and each collected every 30 minutes.

By using files of smaller duration and from the entire period of recording, the processing of the data is much more efficient. The objective is to process the voltage-time series for each file, and using these data, to perform a Fast Fourier Transform (FFT) of the signals to convert them from the time domain to the frequency domain. This procedure allows for the calculation of the Power Spectral Density (PSD) of the signal, which is then used to calculate the Sound Pressure Level (SPL) of the ambient noise. This analysis provides insight into the sound pressure levels generated within the Port, how frequently one sees heavy background noise, and most importantly for this research, at what frequencies the largest amount of ambient sound energy is observed.

Using the data computed from calculating the PSD for each 1.5 sec sample it was possible to generate a spectrogram to illustrate an approximation of how the ambient

noise varies over the 12 hour period of recording. This can be observed in Fig. 2.4.13, with the color bar indicating the amount of energy recorded at varying frequencies.

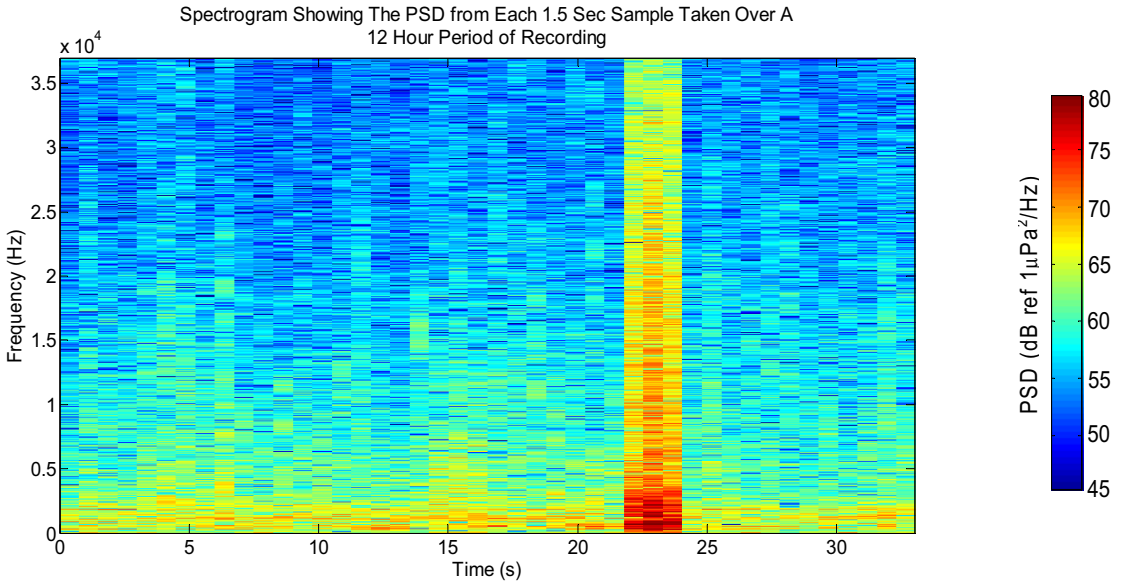


Figure 2.4.13 Spectrogram Showing the Power Spectral Density Over 12 Hour Sampling Period

The spectrogram shown in Fig. 2.4.13 provides a good indication of the frequency band in which one observes the highest values of background noise. It is clear from this chart that consistently through the period of sampling on 26th March 2007 for frequency values below 5 kHz a large amount of background noise is recorded, around a PSD of 75 dB. As previously shown, the Wenz curves [4] indicate that within this frequency region the ambient noise is generated from shipping and industrial activity, and from biological sources. One of the most pervasive sources of biological noise in shallow waters is the snapping shrimp [5]. These animals are common enough in many shallow areas to dominate the ambient noise below 10 kHz [6]. A number of snap samples recorded in the Sydney Harbour, Australia by Ferguson et al. [7] indicate a mean value of 132 dB (ref 1 µPa) SPL, while the work of Whitlow et al. [5] in Kaneohe Bay, Hawaii recorded source levels from 128 dB (ref 1 µPa) to 134 (ref 1 µPa), with the typical spectrum of a click being very broad with only a 20 dB difference between the peak and minimum amplitudes, and having a peak between 2 kHz and 5 kHz, and energy extending out to above 100 kHz. Measurements conducted by Readhead [8] in Gladstone, Queensland, Australia found that snapping shrimp was the dominant source of ambient noise, exceeding that which would be expected from wind generated noise in sea state 7.

Analysis of both the line charts for each sample file, and the spectrogram shown in Fig. 2.4.13 indicates that a large amount of ambient noise was recorded at 1900 on 26th March 2007. The South Turning Notch is frequently used for large commercial ships that employ the use of Port tug vessels when transiting in or out of Port. It is possible that a commercial vessel was present at the time of recording, and combined with the tugs generated this increased amount of ambient noise.

2.4.3.5 Sound Absorption In Turbid Water

An approximation of the absorption of an acoustic signal within the South Turning Notch of Port Everglades has been calculated based on the theory of Fisher and Simmons [9] for absorption of sound in clear seawater, and including contributions due to viscous absorption and scattering from suspended particles [1]. The results are shown in Figs. 2.4.14 and 2.4.15. These plots illustrate the absorption due to clear seawater for the parameters recorded within the South Turning Notch. Also illustrated in the figures is an approximation to the absorption coefficient that includes viscous absorption and scattering from suspended particles in the calculation. As previously discussed, the two acoustic systems that the RPUUV is to employ use a frequency range of 16 kHz – 32 kHz for the remote piloting of the vehicle, and secondly a frequency range of 260 kHz – 380 kHz for the data transmission.

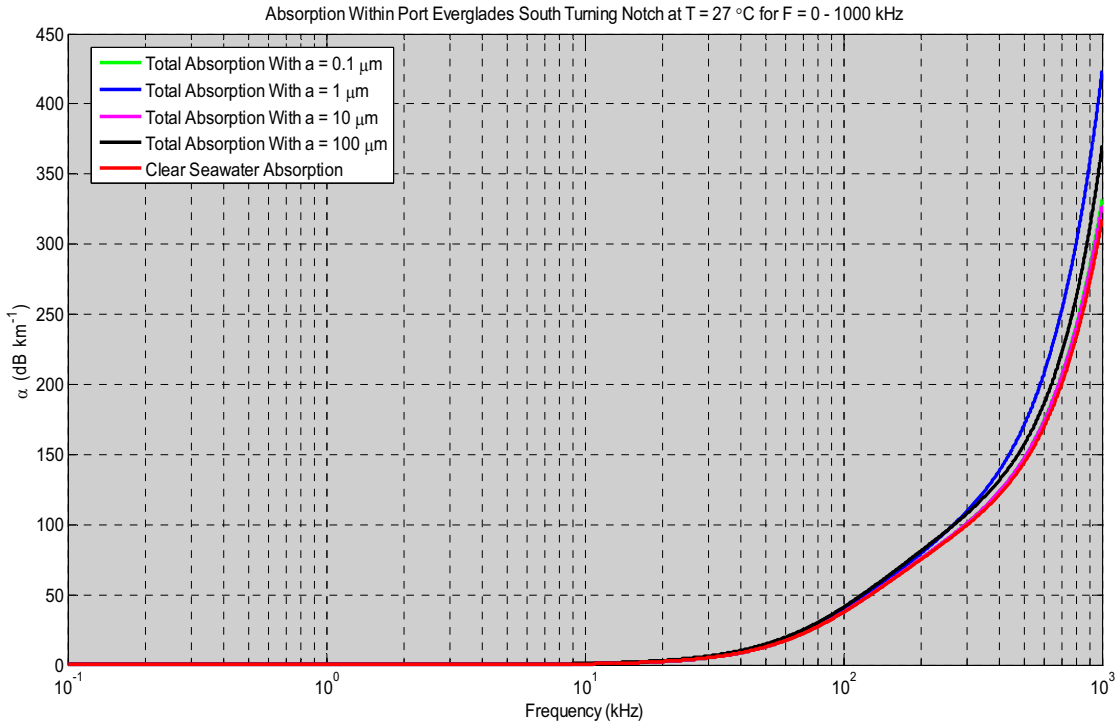


Figure 2.4.14 Total Signal Absorption in Port Everglades South Turning Notch For 0 kHz - 1000 kHz

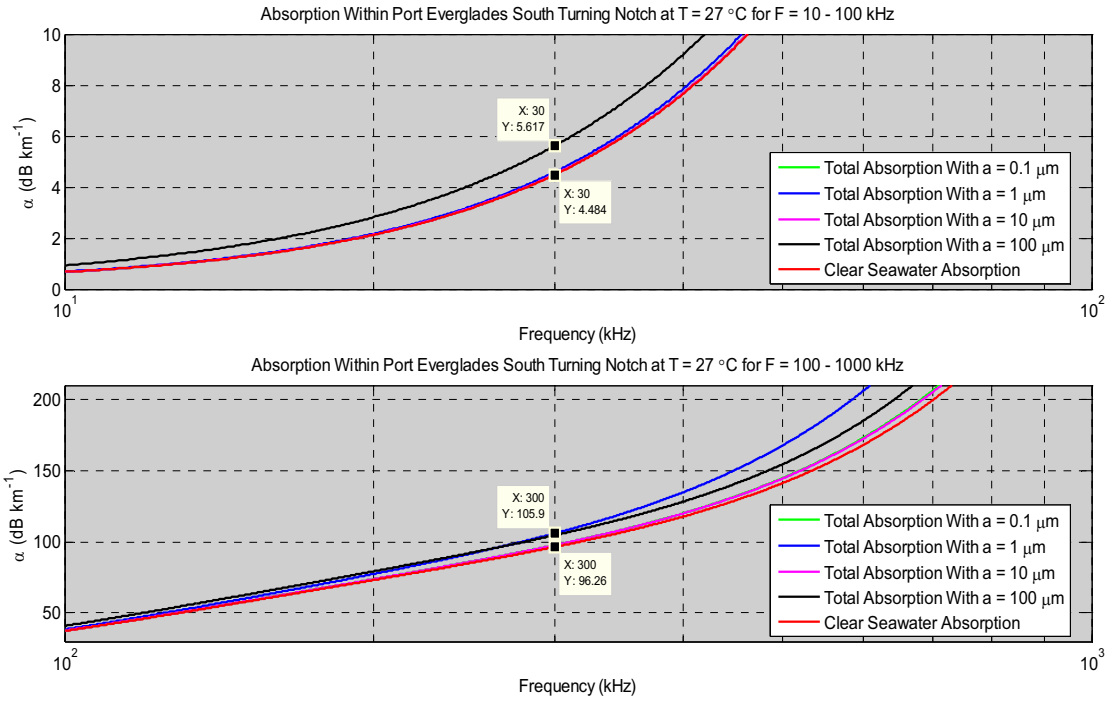


Figure 2.4.15 Total Signal Absorption In Port Everglades South Turning Notch For 0 kHz - 100 kHz And 100 kHz - 1000 kHz

An examination of Figs. 2.4.14 and 2.4.15 shows that for the lower frequencies the absorption of an acoustic signal due to clear seawater is not problematic, with an absorption coefficient of approximately 4.5 dB km^{-1} being exhibited at a frequency of 30 kHz. The results from calculating the absorption coefficient that includes viscous absorption and scattering by suspended particles showed that, for low frequencies and small particle radius, there was negligible impact on the absorption. On the other hand, a large value of particle radius, of the order 100 μm, caused an increase in the absorption coefficient.

As the frequency of the signal increases, however, so does the absorption factor. For clear seawater alone an approximate absorption of 96 dB km^{-1} was observed for a frequency of 300 kHz. The inclusion of viscous absorption and scattering due to suspended particles indicated that for a particle radius of 1 μm and 100 μm the absorption coefficient increased. However, for a particle radius of 0.1 μm and 10 μm the absorption coefficient was the same as that for clear seawater.

The characteristics of the absorption coefficient for the range of values recorded in the South Turning Notch are an important factor to consider when designing a system for this type of environment.

2.4.3.6 Conclusions

The results generated from completing a sampling strategy encompassing 200 different locations within Port Everglades have been detailed within this chapter, with analysis on

how the sound velocity characteristics vary both spatially and temporally throughout the Port.

Analysis of the spatial variation indicated that the sound velocity profile is controlled by the temperature profile, which is a probable reason for the trend of increasing sound velocity with the time of year that was observed from the temporal analysis. The variation observed in salinity had a minimal correlation or impact on the sound velocity, as did the varying tides and currents within the port.

The completion of a more concentrated sampling strategy within the South Turning Notch of Port Everglades, combined with a 12 hour period of ambient noise recording within the same region, has provided insight into the varying conditions that are prevalent in this section. The sound velocity profiles exhibit a slight and linear decrease with depth down to approximately 10 m. Below this depth a more severe drop in sound velocity is observed.

The ambient noise analysis from this area reveals that at lower frequencies, i.e. frequency values below 5 kHz, there is a high level of background activity, which is owing to consistent port and Intracoastal Waterway traffic along with biological sources such as snapping shrimp. It has been highlighted that at 1900 a large amount of energy was recorded. This may be due to increased Port activity within the area at that time, or may be erroneous. One would expect that if it were due to shipping activity, the level of energy would drop off rapidly as the frequency level increases, which is not the case.

Using the data obtained in the turning notch, it was possible to complete a series of absorption calculations that provides insight into the possible degradation of an acoustic signal that may be experienced when operating within this region. These results show that for lower frequency systems, the absorption factor should not pose a problem, but as the frequency of the transmitted signal increases, so too does the absorption parameter.

2.4.4 Optical Characteristics of Port Everglades

Once an initial set of measurements was made, it became obvious that the turbidity characteristics were not constant throughout the port. The amount of data collected during this phase was massive and some condensation of it should prove helpful for its analysis. To aid in the interpretation of the results, the port was divided into four regions shown in Fig. 2.4.16. Region 1 comprises the southern part of the port and includes the Dania cutoff canal. The port section has an average depth of about fourteen meters and the canal has an average depth of four meters. Region 2 is defined as the secure zone and special permission is required to collect data there. Region 3 includes the outlet and the area just inside the port. This area acts as a crossroads and gets the most boat traffic. A high level of mixing from other sections of the port and ocean is expected. Region 4 is defined due to its drastically different bathymetry from the rest of the port with an average depth of 3 meters outside the channel and 8 meters within.

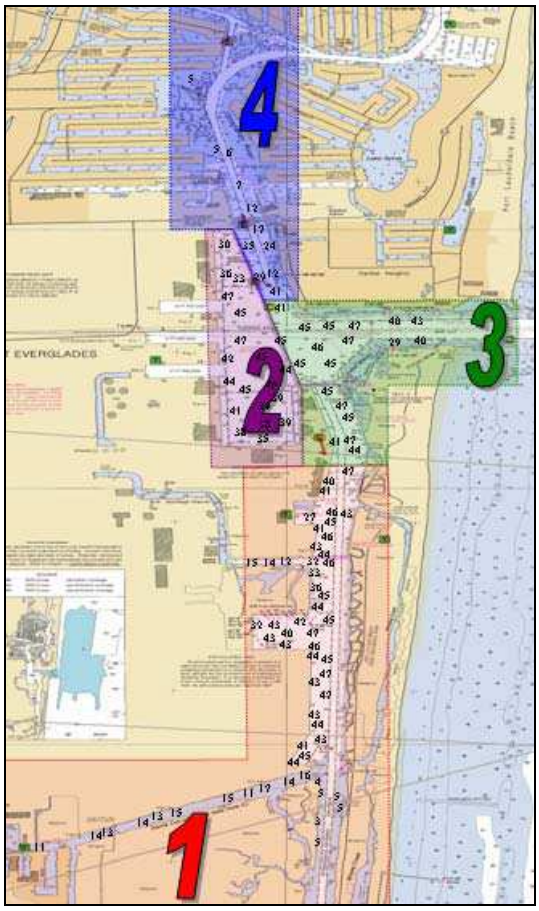


Figure 2.4.16 Port Everglades with 4 Regions shaded in. These regions were selected by comparing turbidity profiles after all measurements had been taken.

From comparing all the graphs of turbidity around the Port, it becomes obvious that certain areas of the port are more prone to high turbidity values. Generally, the turbidity increases southward along the port. This trend is most likely due to decreasing proximity to the port inlet. To test this theory, the proximity of the measurement should be compared to turbidity. When researching this dependence, it became obvious that much of the bulk measurements during the first phase of testing were insufficient for this analysis because of their high variability in location. There are many areas of the port that exhibit localized high turbidity, such as near the shore or within notches along the port that promote stagnation. These areas will have extremely similar proximities to the port inlet but have drastically different turbidity profiles. To negate this variability, the measurements to determine the Inherent Optical Properties (IOP) will be used. These measurements are always taken in the same spot and in deeper water. This approach removes the effects of localized turbidity spikes within the data and gives a more accurate approximation of the proximity dependence on turbidity as can be seen in Fig. 2.4.17. The linear regression curves were calculated by the least squares method within Microsoft Excel.

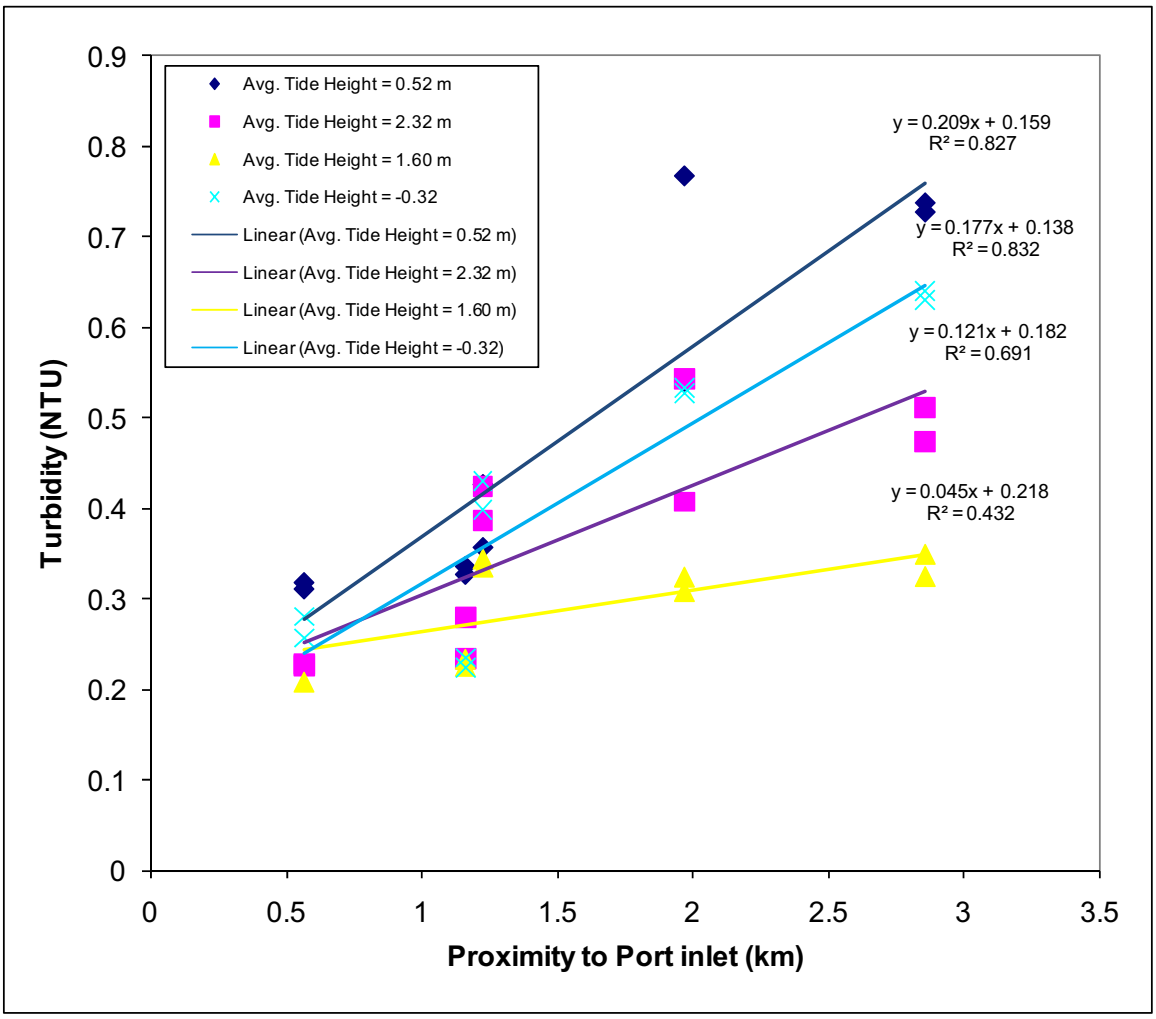


Figure 2.4.17 Proximity to the Port inlet vs. Turbidity. Each data set is from a different day and the average tide height during the measurements is given.

Each day was plotted as a different data set which shows distinctly different slopes. As can be seen from spatial plots of the port, the turbidity can rise throughout the port which would usually be associated with the tide. To check if the tides control the slope seen on this graph, average tide height vs. turbidity gradient was plotted in Fig. 2.4.18.

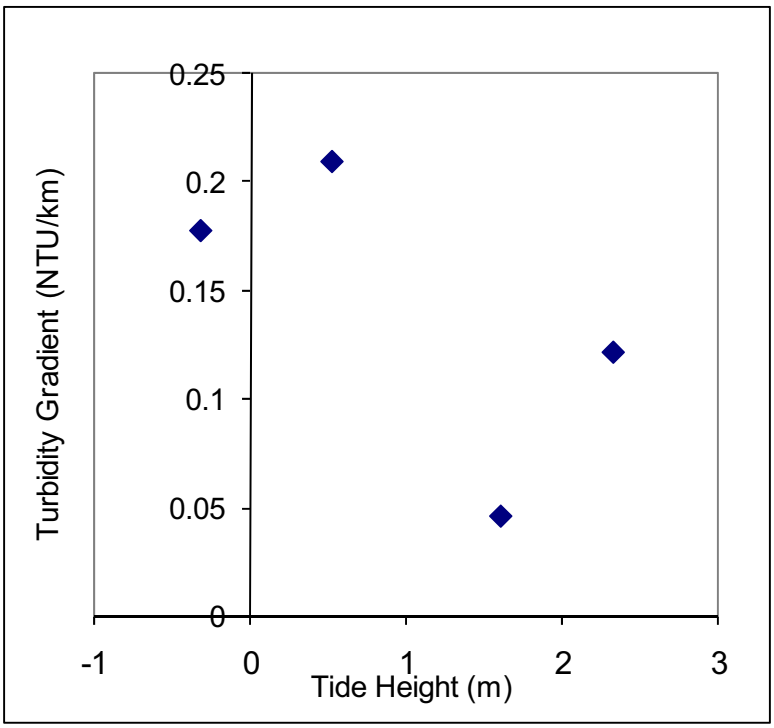


Figure 2.4.18 Slope vs. Tide height. Slope is determined from the previous plot of Proximity vs. Turbidity.

However, this plot does not show a good connection between tide height and the gradient from the previous plot. This result weakens this hypothesis, but additional measurements are necessary to draw a concrete conclusion.

2.4.4.1 Correlations between Inherent Optical Properties (IOP) and Turbidity

Turbidity values, as previously mentioned, are not an IOP of the water. Instead, they are estimates of visibility that are dependent on the specific water being measured. By comparing the turbidity values taken simultaneous to the IOP measurements, a correlation between the two can be surmised. This will add pertinence to the turbidity values taken during the first sampling period in which only turbidity values were taken to define the port's optical properties. These correlations can also be used for future measurements taken within the port using a turbidity meter.

Due to the nature of the turbidity measurements, turbidity values should most closely coincide with the scattering coefficient. The scattering coefficient used for this correlation was derived from the difference of attenuation and absorption at 535 nm (cf. Fig. 2.4.19).

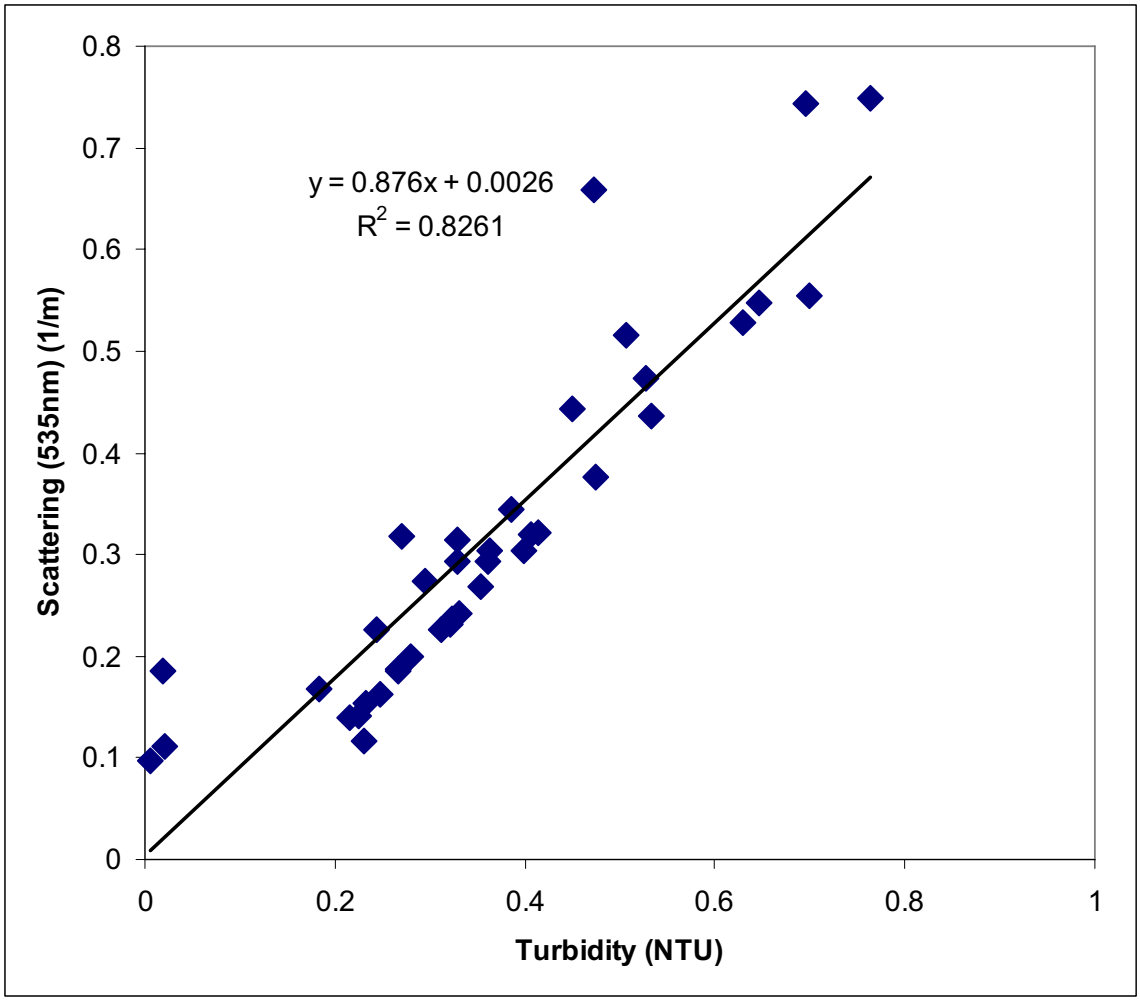


Figure 2.4.19 Turbidity vs. Scattering at 535 nm.

As the plot shows, there is a reasonable correlation between turbidity and the scattering coefficient for these waters. Since turbidity meters measure the reflectance of the water, it would be reasonable to assume that water with zero scatterance would result in a zero turbidity value. This is reasonably indicated by the regression line's 0.0026 y-axis intercept. There are three values that show almost zero turbidity but still register a non-zero scattering coefficient. This is most likely due to the highly peaked phase scattering function of pure water in the forward direction. Scattering in the forward direction would not register on a turbidity meter, but would show an increase in attenuance on an attenuation meter.

Within a specific body of water, a relationship between turbidity and absorption usually exists (cf. Fig. 2.4.20). This relationship is not based on the instrumentation, as a turbidity meter can not directly measure absorption, but is rather based on relative constituent concentrations. Chromophoric Dissolved Organic Matter (CDOM) is highly absorbing and can be increased by the re-suspension of sediments [10]. The increased sediment concentration will yield a higher turbidity value, while the CDOM concentration will yield a higher absorption value.

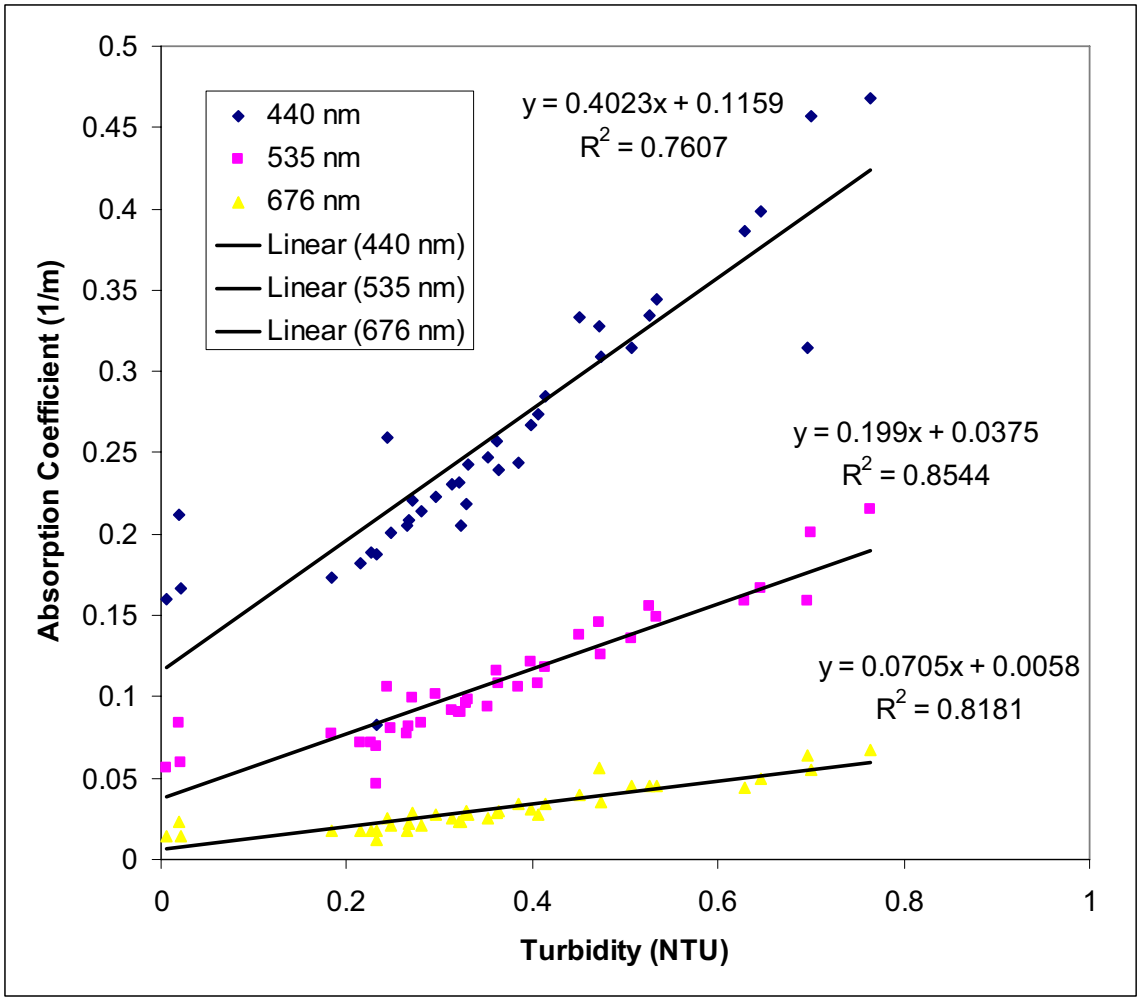


Figure 2.4.20 Turbidity vs. Constituent Absorption at three wavelengths, 440 nm, 535 nm, and 676 nm

2.4.4.2 Visibility Approximations

A relationship between turbidity and visibility [2] is shown in Fig. 2.4.21. Object recognition by a multi-wavelength detector of the presence of a dark shape (black disk) will first become apparent at the first available wavelengths. But to distinguish between wavelengths the attenuation at that specific wavelength needs to be sufficiently low. Thus the top curve should give an indication of the visible range for detection of an objects presence whereas the bottom curve should give an indication of the range at which an object's full visible spectral properties can be observed.

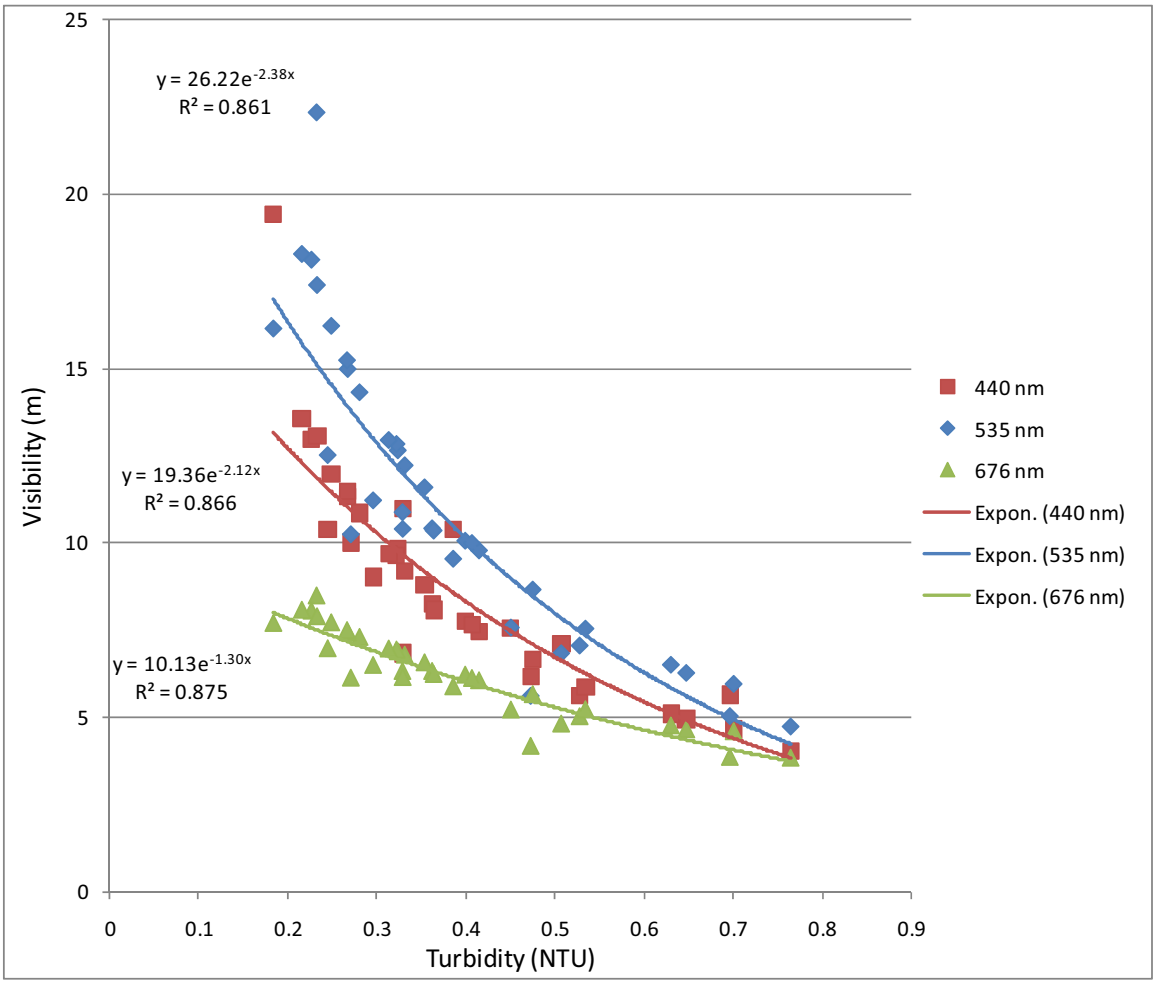


Figure 2.4.21 Visibility vs. Turbidity

2.4.4.3 Conclusions

The optical properties within Port Everglades are subject to high variability. Turbidity values range between 0.2 and 1.2 NTU and show a general decrease with proximity to the Port inlet and only a weak correlation to the tidal level. The turbidity profiles generally show an increase at greater depth due to the settling characteristics of stirred up sediment. Spectral absorption curves exhibit a strong likeness to exponential curves with negative slopes indicative of high concentrations of CDOM. Spectral attenuation ranged between 0.3 and 1.3 m^{-1} with generally wavelength independent contributions from scattering. Turbidity showed a high correlation to the scattering at 535nm, though the proportionality constant between the two is very similar at all wavelengths. Turbidity also showed high wavelength dependence to the absorption coefficient. The proportionality constants found here can be applied to future turbidity measurements to estimate other inherent optical properties within Port Everglades.

Using the values of attenuation within the Port along with a rough estimation as to its visibility implications [2], the visibility range of the RPUUV should range between 16 m

and 3.7 m. This range should be considered with caution as it assumes that the target is a black disk viewed horizontally in full daylight directly under the surface. In the Port waters this range will change with depth, solar zenith angle, cloud cover, artificial illumination and shadows. This work has focused on providing the tools necessary to increase this range using clever camera and lighting optimizations. Further advances in the turbid water imaging would require fundamental changes by employing such systems as range gate imaging or laser line scanning.

References for Section 2.4

[1] Sheahan, D., "An Investigation into the Acoustic Variability and the Attenuation of an Acoustic Signal within a Port Environment Focusing on Port Everglades, Florida," M.S. Thesis, Florida Atlantic University, Boca Raton, FL *August 2007*.
[2] Whipple, D., "Optical Characterization of Port Everglades Focusing on Underwater Visibility," M.S. Thesis, Florida Atlantic University, Boca Raton, FL *August 2007*.
[3] Captain Voyager Charting Software, Version 5.3.2, Nautical Technologies Ltd.
[4] Wenz, G. M., 'Acoustic Ambient Noise in the Ocean: Spectra and Sources', *The Journal of the Acoustical Society of America*, 134(12): 1936 – 1957, (1962).
[5] Whitlow, W. L., Banks, K., 'The Acoustics of the Snapping Shrimp *Synalpheus Parneomeris* in Kaneohe Bay', *The Journal of the Acoustical Society of America*, 103(1): 41 – 47, (1998).
[6] Tyack, P. L., Howald, T., 'Biological Sources of Noise in Coastal Waters', *The Journal of the Acoustical Society of America*, 94(3): 1819, (1993)
[7] Ferguson, B. G., Cleary, J. L., 'In-Situ Source Level and Source Position Estimates of Biological Transient Signals Produced by Snapping Shrimp in an Underwater Environment', *The Journal of the Acoustical Society of America*, 109(6): 3031 – 3037, (2001).
[8] Readhead, M. L., 'Snapping Shrimp Noise Near Gladstone, Queensland', *The Journal of the Acoustical Society of America*, 101(3): 1718 – 1722, (1997).
[9] Fisher, F. H., and Simmons, V. P., "Sound Absorption in Sea Water", *The Journal of the Acoustical Society of America*, 62: 558 – 564, (1977).
[10] Bukata, R. P., J. H. Jerome, K. Y. Kondratyev, and D. V. Pozdnyakov, *Optical Properties and Remote Sensing of Inland and Coastal Waters*, 362 pp., CR press, Boca Raton, FL (1995).

2.5 Development of a High Resolution Imaging Sonar for Underwater Inspections
PI: Dr. S. Schock

Tasks 3.13 - 3.17

2.5.1 Summary

A high resolution focusing sidelooking sonar (“ acoustic camera”) was developed to generate near photographic-like images of underwater objects from maneuvering underwater vehicles. The sonar has a hemispherical acoustic projector and a 512 hydrophone line array. The resolution of the sonar is 1.5 mm in the nearfield for a center frequency of 1.6 MHz and 400 kHz of bandwidth. The resolution is substantially better (by more than a factor of 2) than commercially available sonars. Tests in the vicinity of Port Everglades, Florida demonstrated the capability of the sonar for imaging ship hulls in water with high turbidity (poor visibility). During port tests, the acoustic camera was mounted in the RPUUV which performed hull surveys and transmitted image data to the topside display computer via the RF modem. A significant result is that the acoustic camera generated high resolution imagery with a 90 degree field of view and 1.5 mm resolution while the UUV was maneuvering. Images of underwater zincs anodes mounted to ship hulls demonstrate that the sonar is capable of imaging WMD attached to ship hulls. This new acoustic imaging technology is an important development because the images produced by the acoustic camera have photographic-like resolution that allows the operator to easily recognize WMD mounted on seawalls or hulls in turbid water where visual and optical searches are not possible.

2.5.2 Sonar Design and Construction

The acoustic camera design allowed the sonar to mount on the side of the RPUUV (remotely piloted unmanned underwater vehicle) for the purpose of conducting port inspections. Figure 2.5.1 shows a photograph of the acoustic camera mounted on the side of the RPUUV.

The array can be rotated around a horizontal axis passing through the array elements. This adjustment allows the operator to set the elevation angle of the array to minimize effects of seasurface or seabed scattering and to obtain the best geometry for imaging with the 2D camera.

The acoustic array housing shown in Figure 2.5.1 contains a line array of 512 rectangular 1-3 composite hydrophones with dimensions of 0.6 by 1.2 mm and a element spacing of 1mm. Figure 2.5.1 also shows the hemispherical projector which is used to illuminate a wide field of view with a single transmission. A switching amplifier which drives the projector is mounted in the small housing behind the projector.

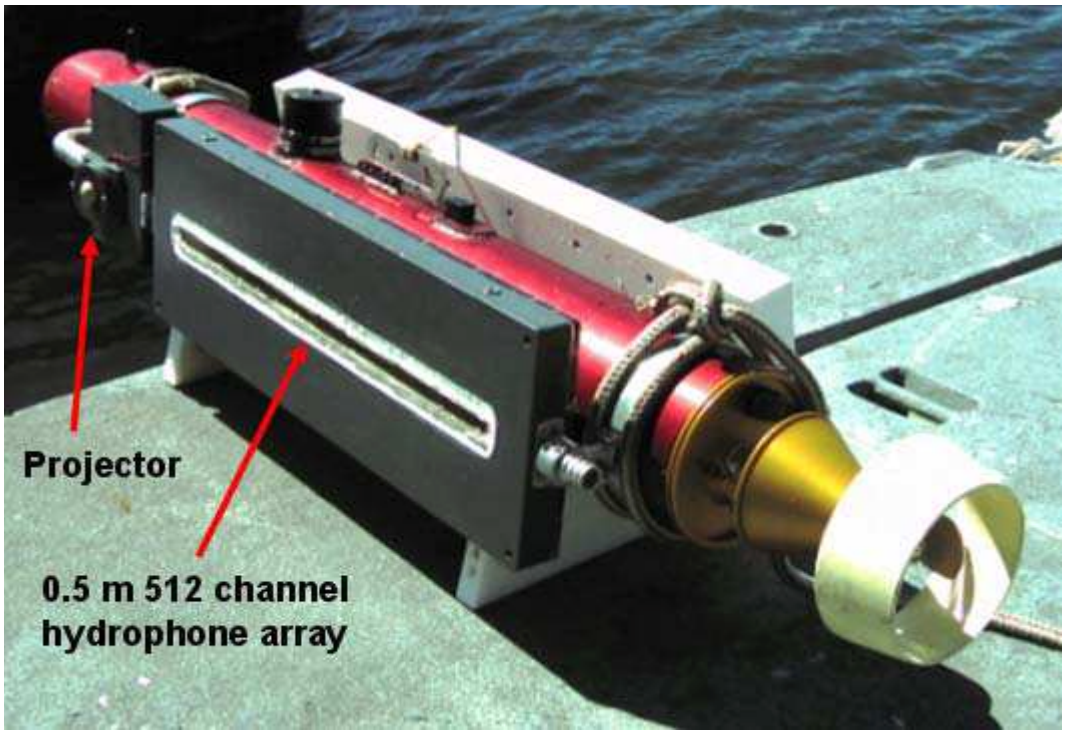


Figure 2.5.1. Photograph of RPUUV with side-mounted acoustic camera

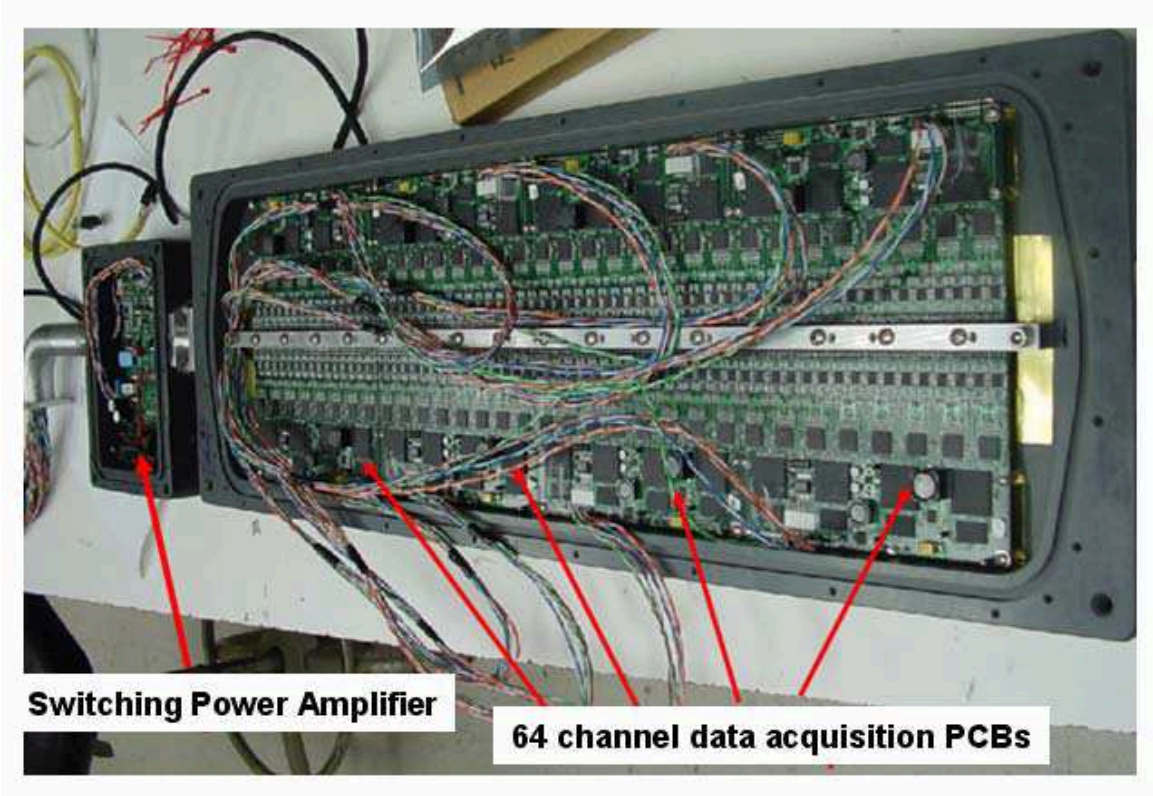


Figure 2.5.2 Back side of acoustic array housing shown in Figure 2.5.1. The hydrophone array housing contains eight 64 channel data acquisition cards. The smaller housing on the left contains the switching power amplifier used to drive the hemispherical projector.

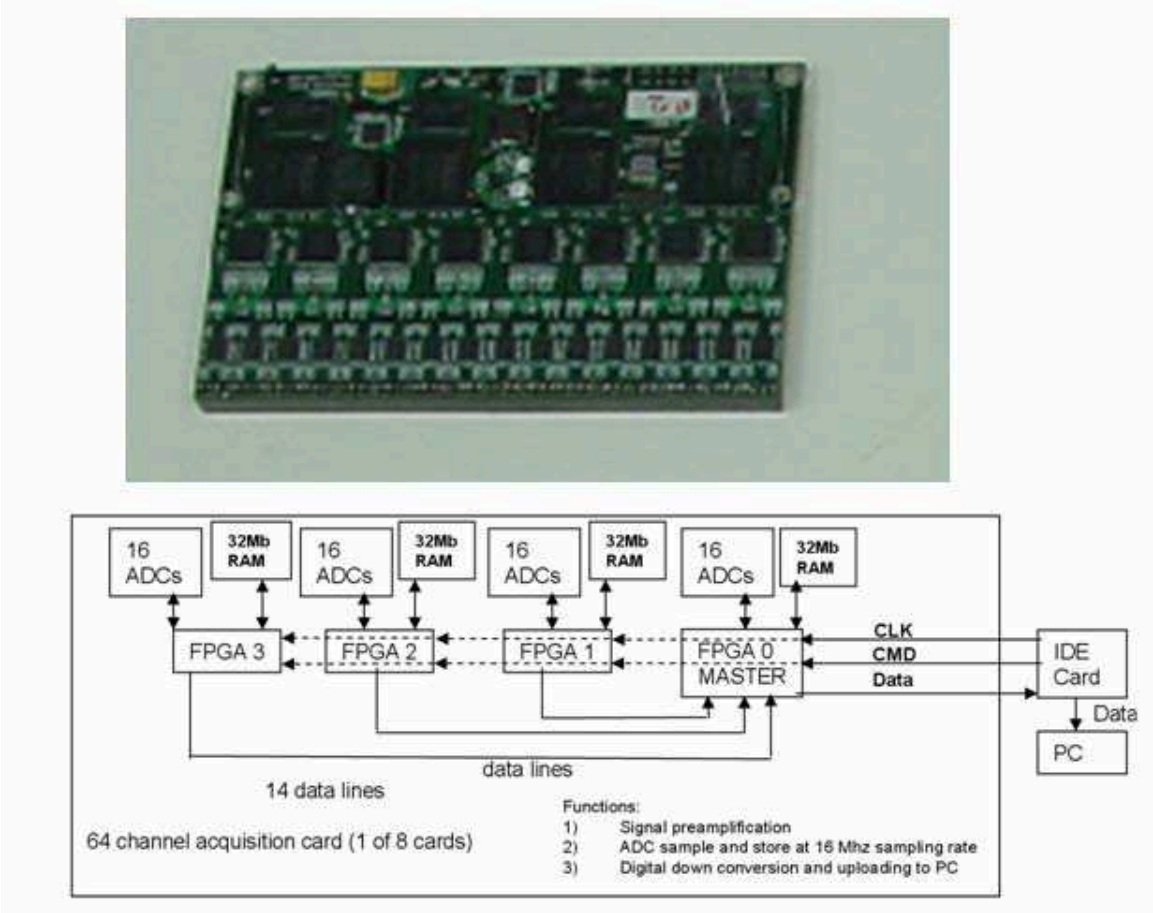


Figure 2.5.3 64 channel data acquisition card photo and data flow

The acquisition and processing steps are the following:

1. During data acquisition each FPGA acquires data for 20 msec at a rate of 480 Mbytes sec. (15 Gbytes per second burst rate) and stores the data in a 32 Mbyte DRAM.
2. In between the data acquisition cycles, the FPGA reads the DRAM and downconverts the data to baseband.
   For the sea testd conducted in this report, the ADC sampling rate was 16 Msamples/sec. The data was downconverted by 32 to 1 to yield a 520 kHz baseband bandwidth. (The basebanded complex sampling rate is 520 kHz)
3. The baseband data is transferred to the PC via the IDE interface card. The FPGA acts as a multiplexer that streams the 64 channels of data into a serial data stream to the PC.
4. The PC either stores the downconverted data or processes the data to form an image.
5. The data processing steps are the following:
   a. Matched filtering time compresses the data to obtain temporal resolution of 1.5 mm. During the matched filter processing, the forward FFT size is 8k and the inverse FFT size is 128k.

b. Nearfield focusing corrects for two spherical spreading and coherently sums the data at each focal point in a plane to form a 2D image of a 3D scene. The interval between focal points is normally set to 0.5 mm.

6. Ethernet is used to communicate with the Host UUV PC and to upload imagery or downconverted raw data via the RF link.

The PC and power conditioning electronics were packaged into the hull of the RPUUV. Figure 2.5.4 is a photo of the acoustic camera electronics package just before it was inserted into the RPUUV housing.

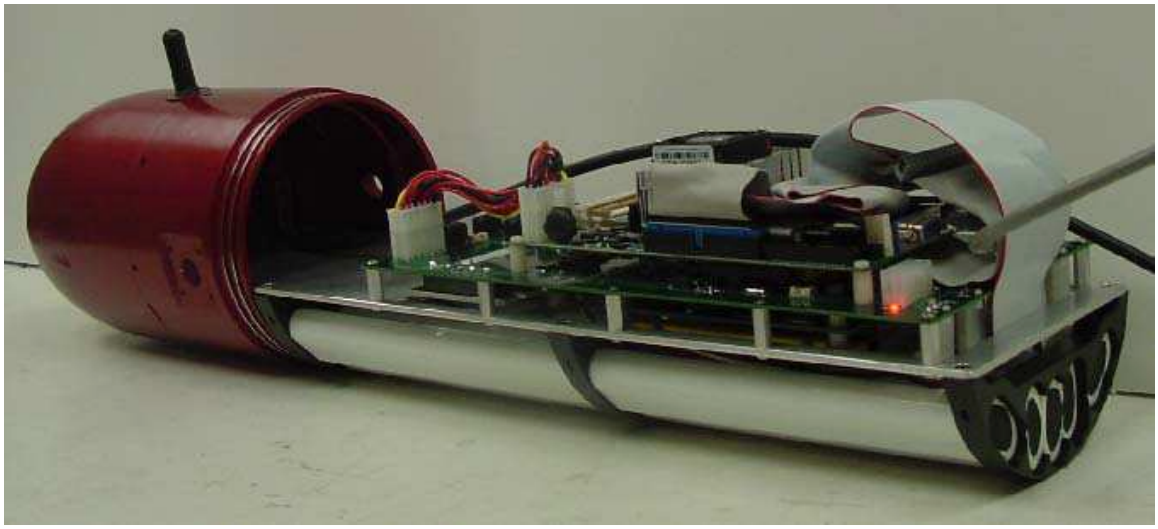


Figure 2.5.4. Acoustic camera electronics mounted in the RPUUV includes a PC processor, IDE interface PCB, power supply PCB and batteries.

2.5.3 Tests of Acoustic Camera prior to RPUUV operations

Tank tests were conducted to verify that the acoustic camera met the predicted resolution performance. Scattering off the surface of a 3/8" stainless steel ball, generated by a transmitting a 1.4 to 1.8 MHz pulse, was processed using nearfield focusing to form a focused echo. The focused echo off the 3/8" diameter stainless steel ball had a measured range resolution of 1.7mm and an azimuthal resolution of 1.5 mm at a range of 0.9 m which agrees with the 1.7 by 1.7mm of the simulator used to develop the acoustic camera design. The simulated and actual focused echoes for an effective point source are shown in Figure 2.5.5.

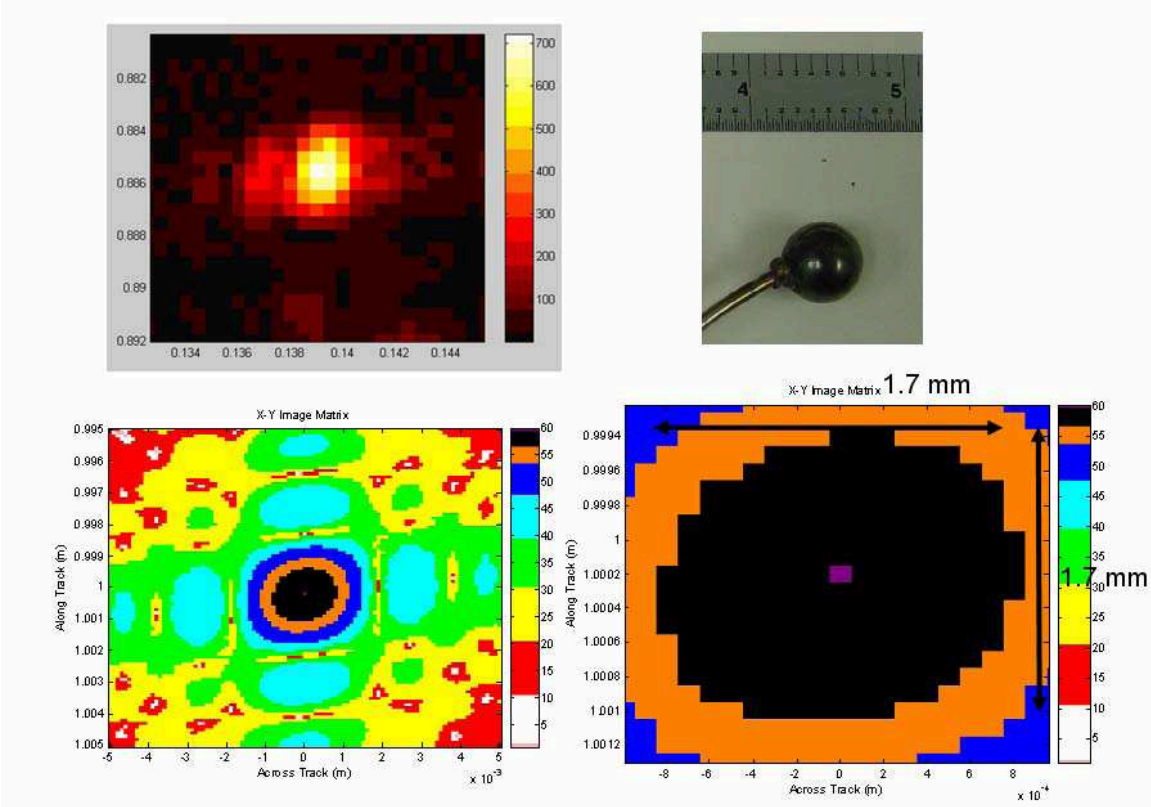


Figure 2.5.5 Verification of 1.5 mm resolution in the near field. The focused echo (top left) off a 3/8" diameter stainless steel ball (top right) agrees with resolution of acoustic camera simulated imagery (bottom). The scale of the acoustic image is in meters.

The performance of the acoustic camera was tested in low visibility harbor water by comparing optical camera imagery and acoustic camera imagery of an object attached to the bottom of a hull. The visibility of the harbor water was approximately 1 meter. Figure 2.5.6 shows an optical image and an acoustic image of a 30 cm long zinc anode attached to a hull. Both images were taken at a range of 80 cm which is near the maximum range of optical underwater visibility at the time of the measurements. All images were generated at a center frequency of 1.6 MHz using 400 kHz of acoustic bandwidth. Pulse length was 0.25 msec.

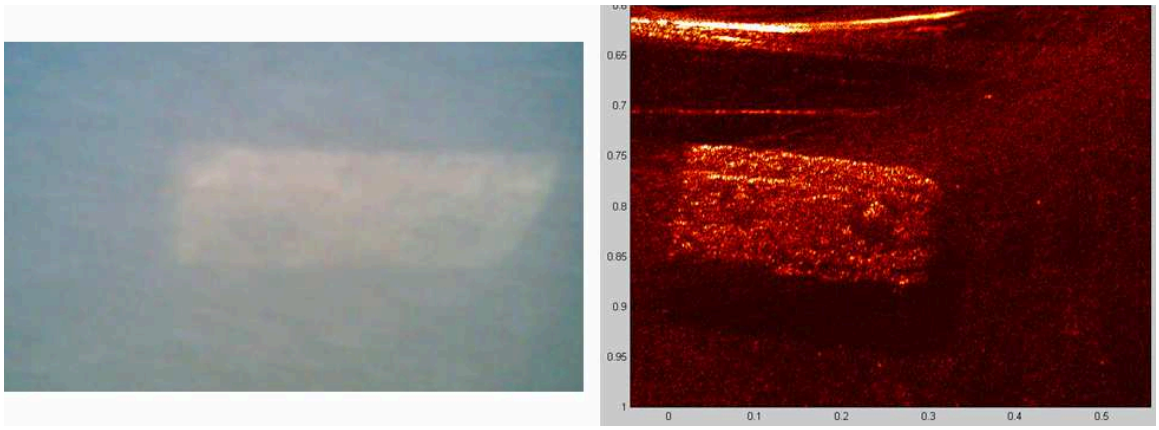
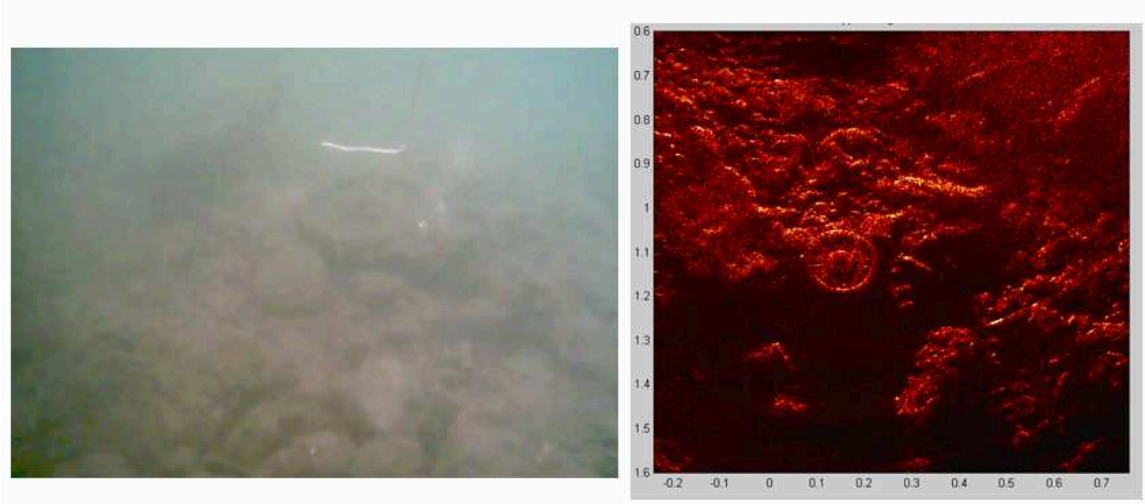


Figure 2.5.6 Comparison of optical image and acoustic camera image of a 30 cm hull mounted zinc anode. Water visibility was approximately 1 meter. The scale of acoustic image is in meters

An image of a small object resting on seafloor rubble was made to determine the effectiveness of locating and identifying small objects on harbor floor in low visibility water. Figure 2.5.7 provides a comparison between an optical and acoustic camera images of a 15 cm diameter barbell weight resting on the seabed in Port Everglades. The seabed is covered by coral fragments.



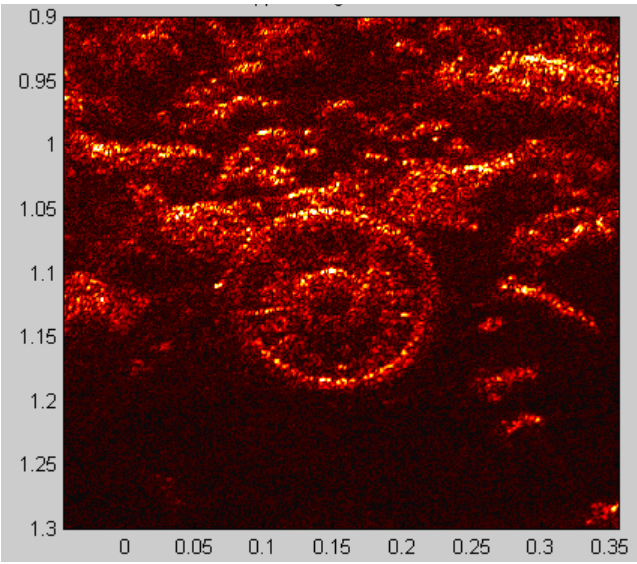


Figure 2.5.7 Optical image (upper left) of a 15 cm diameter barbell weight resting on coral debris in Port Everglades. The optical and acoustic images were taken at the same time from identical locations. The lower acoustic image is a zoomed version of the upper right acoustic image of the seabed. The 1.5 mm resolution camera almost resolves the lettering on the barbell weight. The scale of the acoustic images is in meters.

The acoustic camera generated images with about a 90 degree field of view which is about 3 times wider than the leading commercial acoustic camera widely used by the US Navy. An example of an image showing at least a 90 degree field of view is given in Figure 2.5.8.

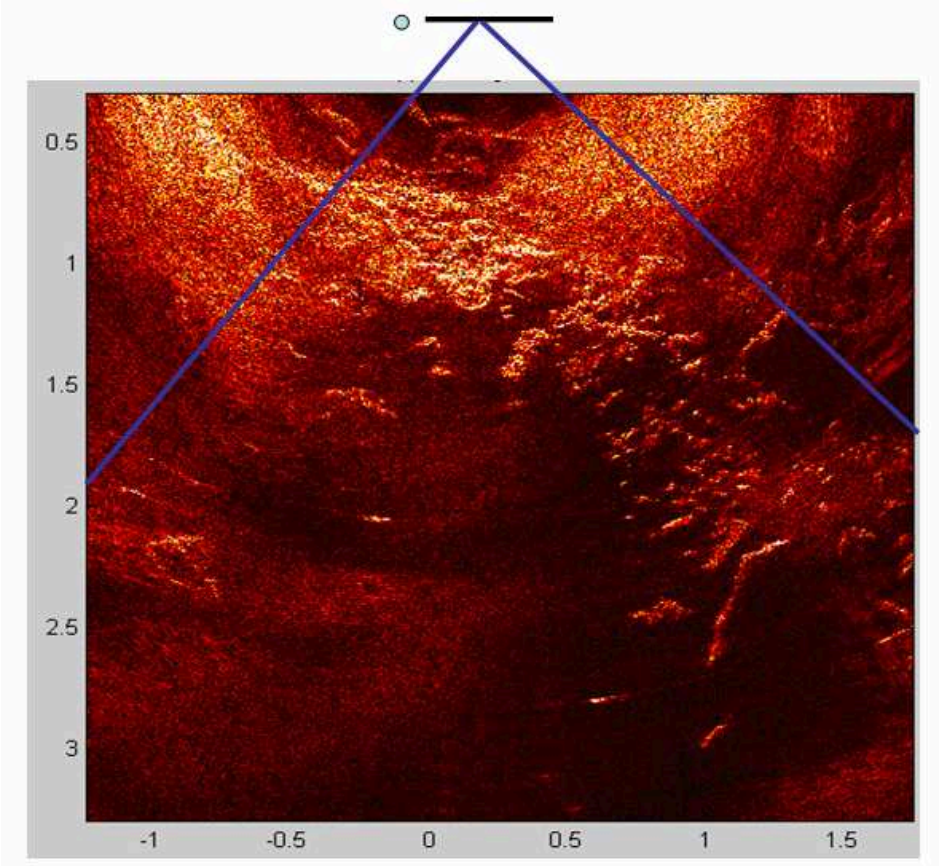


Figure 2.5.8 Image showing a 90 degree wide field of view.

2.5.4 Tests of RPUUV in Harbor Waters

The acoustic camera array was mounted with a vertical orientation for the first set of tests so the RPUUV could use the acoustic camera to scan the underside of a hull. Acoustic camera image data were generated at 1 sec intervals as the RPUUV ran alongside the hull of the R/V Stephan. The water depth was approximately 4 meters. During the tests, the horizontal offset between the RPUUV and the ship's hull was varied between 2 and 8 feet for RPUUV depths of 24 and 52 inches. The Figure 2.5.9 shows the RPUUV surface float and RF modem float during a hull survey of R/V Stephan. The image of a zinc anode in Figure 2.5.10 was produced during the R/V Stephan hull survey.

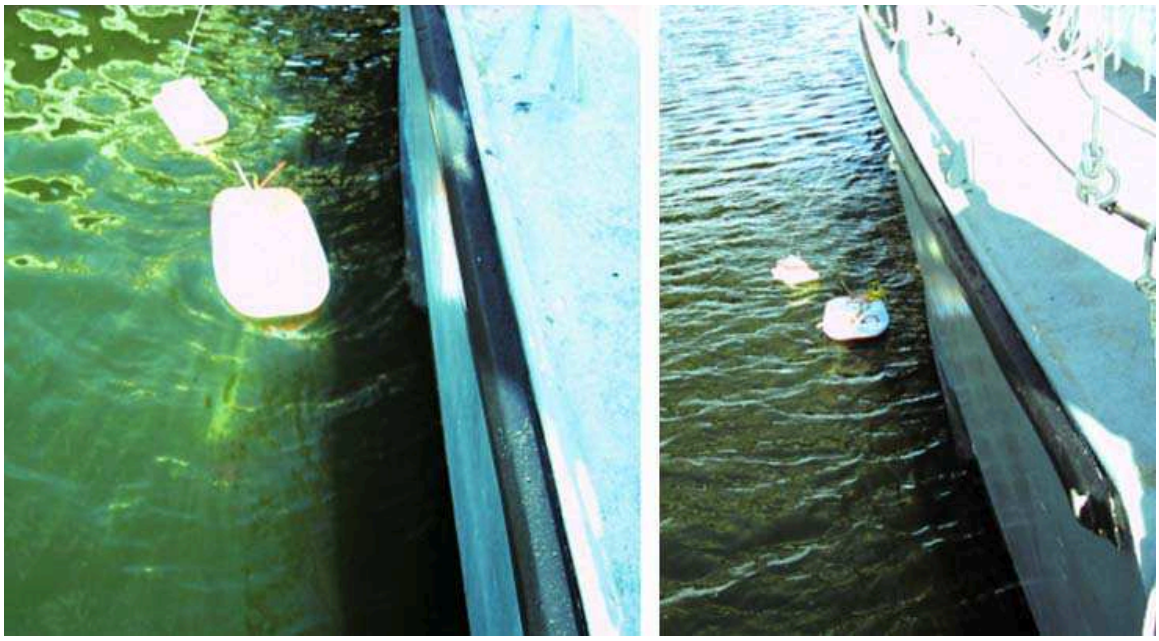


Figure 2.5.9 RPUUV operations along the hull of the R/V Stephan

During the sea trials the sonar communications interface was also tested. The sonar was controlled via the RF ethernet link and acoustic data was retrieved from the RPUUV via the RF ethernet link.

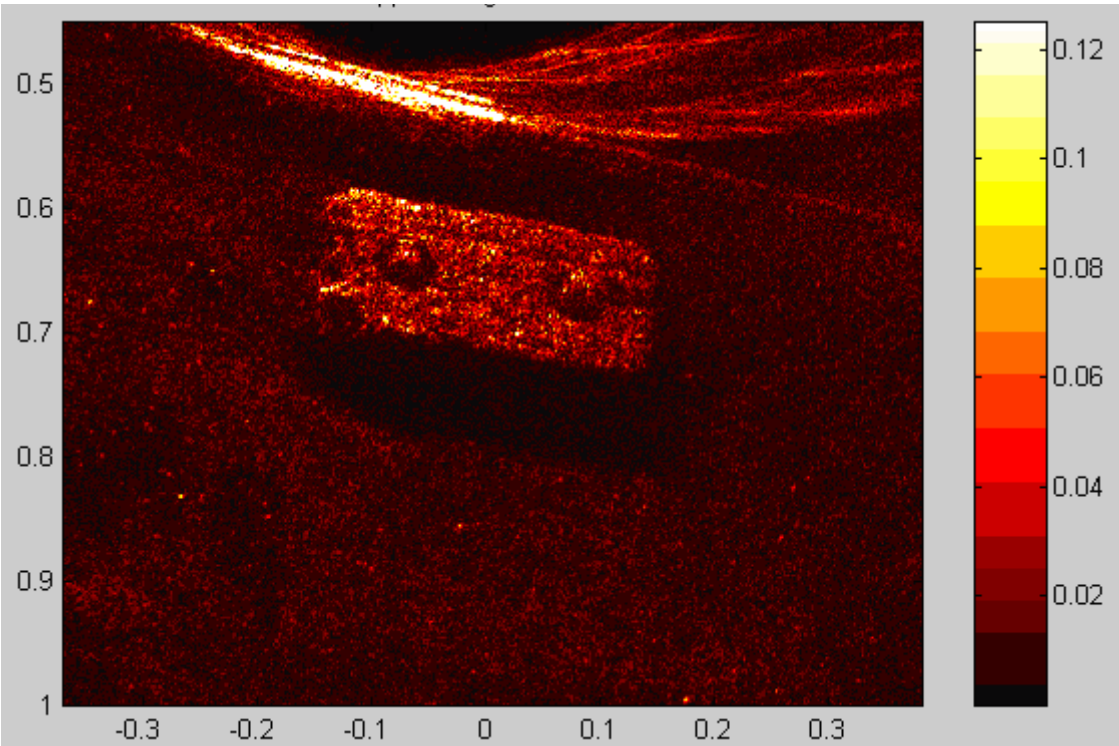


Figure 2.5.10 Image of 30 cm long zinc anode mounted on hull of RV Stephan. This image was generated during a RPUUV hull survey of R/V Stephan.

Harbor testing of the RPUUV also included a survey of pilings. The camera's hydrophone array housing was tilted downward to an angle of 45 degrees from vertical. This array elevation angle prevented sea surface scattering noise and provided a good geometry for imaging an upright object. An example of a downward looking acoustic image of a piling is given in Figure 2.5.11.

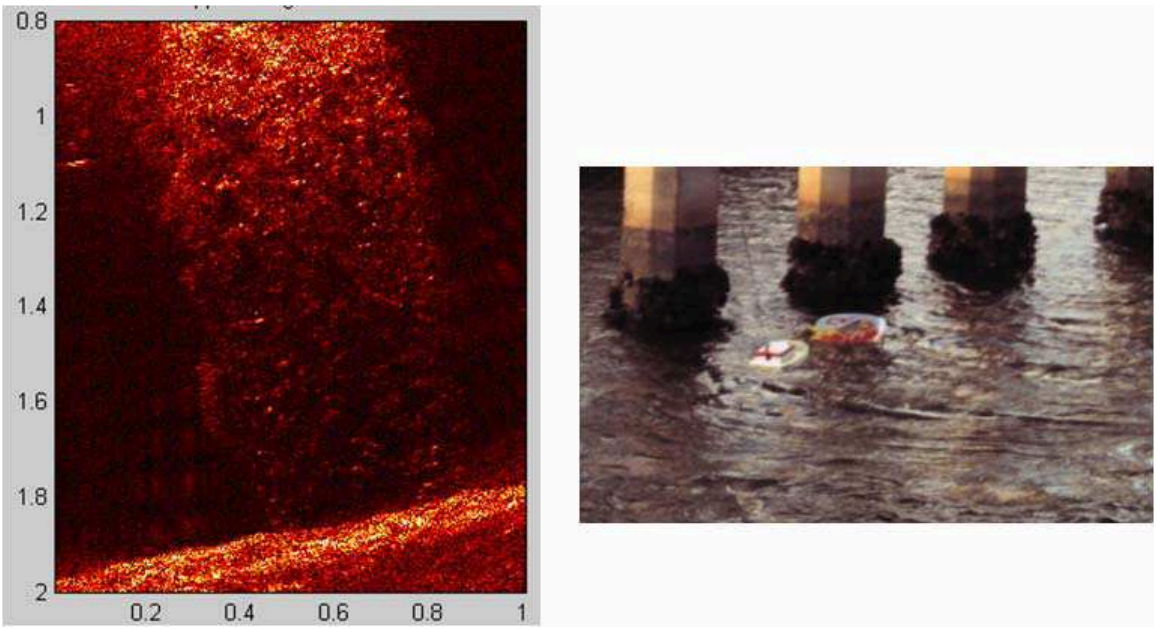


Figure 2.5.11. During RPUUV operations, the acoustic camera generated an image of a concrete piling encrusted with marine growth. The seabed is at the bottom of the image. The image scale is in meters.

2.5.5 Conclusions

Acoustic tank and harbor tests of the acoustic camera demonstrated that the acoustic camera is a very capable tool for imaging in turbid coastal waters. Testing showed that 1.5 mm temporal and azimuthal resolutions provided near optical quality imagery. It is possible that the newly developed acoustic camera is the highest resolution acoustic camera in existence for the application of underwater imaging..

A summary of camera features:

1) The modular array design allows expansion of the hydrophone array to extend the range of nearfield imaging. For example a 2 meter, 4096 element array would extend the 1.5 by 1.5 mm resolution to a range of about 4 meters.
2) Wide field of view (90 degrees at 1.6 MHz). The field of view is even wider at lower operating frequencies.
3) Images are formed using a single transmission thereby providing the capability of generating high resolution images from maneuvering or unstable underwater vehicles without the need for motion compensation.
4) The general purpose hardware and software design allows for interchangeable projectors so that the operating frequency and bandwidth of the sonar can be changed by swapping out projectors. Using a projector with wider bandwidth and high operating frequencies would provide higher resolution images. Sub-millimeter resolution is attainable with a small level of effort.

Tests showed that the camera can be deployed on underwater vehicles for underwater inspections of harbor and ship hulls. This camera is expected to have a wide range of defense applications including mine identification in coastal waters and homeland security applications such as scanning ship hulls for WMD or contraband.

Hydrodynamic refinement and characterization of a small underwater vehicle for hull and harbor survey
PI: Karl von Ellenrieder

Tasks 3.18-3.21

2.6.1 Summary

The objectives of this work were to refine the hydrodynamic design and more fully characterize the hydrodynamic/dynamic coefficients of the RPUUV as well as to investigate the control surface (propeller duct) configuration. This report describes the RPUUV model, the laboratory setup, and the experimentally determined hydrodynamic/dynamic coefficients. One unexpected development in the evolution of the vehicle was the addition of a set of fairly large sonar array panels. The hydrodynamic performance of the vehicle with these panels was also investigated.

Some of the key findings and future design recommendations include: 1) the propeller duct and its supporting structure contribute significantly to the overall drag of the vehicle. Modification of the strut support cross section should mitigate this effect. 2) In a turn the propeller duct produces a beneficial turning moment that counteracts the Munk moment hull. If maneuverability is found to be an issue, increasing the chord length of the propeller duct slightly could help to improve the vehicle's controllability. 3) The new sonar panel design introduces both a substantial drag penalty (C_d is roughly doubled) and a large sway force generated when the vehicle is in a turn. If the controllability of the vehicle in turns is found to be an issue, an increase in the propeller duct length may be required.

2.6.2 Introduction

The following tasks were performed during Year 3:

[a] Task 3.18: Construction of experimental models and modifications of flow facility mounting supports.

[b] Task 3.19: Experimental determination of the hydrodynamic characteristics and dynamics coefficients of the model hull form in a 4'x4' towing tank.

[c] Task 3.20: Testing of different control surface configurations to optimize the design.

[d] Task 3.21: Experimental data, an evaluation of the hydrodynamic design and recommendations for future innovations are included in this year 3 final report.

2.6.2-1 Background

Previous experimental hydrodynamic work on the RPUUV involved the design and testing of the system, with most emphasis placed on characterizing the thrust and moment produced by the vectored tail thruster as a function of vehicle yaw and thruster rudder angle. Here, the previous results are extended through a suite of force/torque transducer measurements to further examine the hydrodynamic coefficients (lift, drag and moment) of the RPUUV hull at higher speeds and to explore the hydrodynamic effects of the propeller duct (vehicle control surface) configuration. In addition, an alternate configuration of the vehicle, in which the sonar system is externally mounted from the vehicle rather than contained within the vehicle's hull, has been tested. In the process, modifications to the towing carriage and flow facility mounting supports, as well as construction of model components were performed.

Experiments were carried out on a 1:1 scale model of the vectored-thruster RPUUV (Figure 2.6.1). The hull of the model has a length of l_v = 0.914 m and a diameter of d_v = 15.24 cm. The vectored thruster consists of a ducted propeller mounted on a gimbaled motor such that the propeller axis can trace out a cone with a half-angle of about 40^o. A Wageningen (MARIN) 19A circular duct [4] (also see Figure 2.6.2) with a chord-to-diameter ratio of c/D = 0.5 is used in combination with a D = 13.2 cm diameter, A-type three-bladed propeller. The propeller has a pitch/diameter ratio of 1.02 and is designed to operate at an advance ratio of J = 0.31. The propeller has been removed from the model for the tests described in this report.

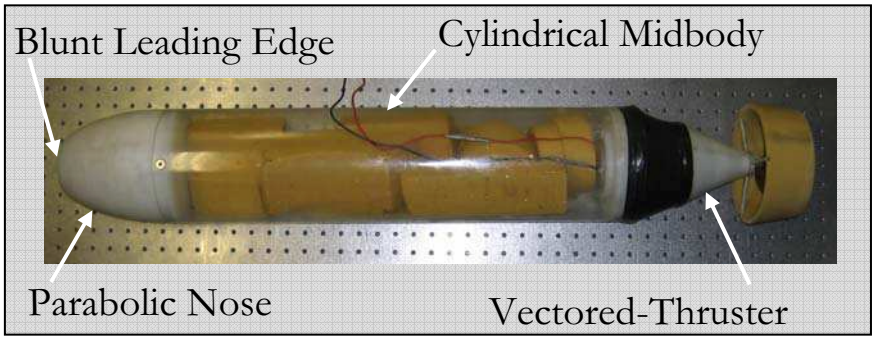


Figure 2.6.1: The RPUUV Model.

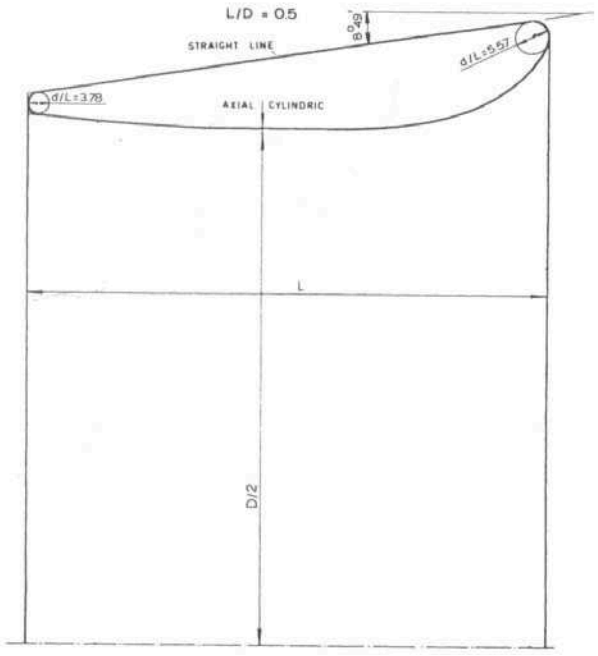


Figure 2.6.2: Wageningen Model 19A duct profile.

2.6.3 Experimental Setup

Experiments were conducted in the 0.6x1.22x10.7 [m^3] test section of the FAU SeaTech towing tank at the conditions indicated in Table 2.6.1. Here the Reynolds number is defined as $R = Ul_v/\nu$ and the Froude number is $F = U/(gl_v)^{1/2}$, where U is the towing speed, ν is the kinematic viscosity of freshwater at 20°C and g = 9.81 [m/s^2] is the gravitational constant.

Table 2.6.1: Test Conditions.

Case	Towing Speed [knots]	Reynolds Number	Froude Number
1	$U_1 = 0.3$	$R_1 = 1.4 \times 10^5$	$F_1 = 0.05$
2	$U_2 = 0.6$	$R_2 = 2.8 \times 10^5$	$F_2 = 0.10$
3	$U_3 = 0.5$	$R_3 = 4.7 \times 10^5$	$F_3 = 0.17$
4	$U_4 = 1.5$	$R_4 = 7.1 \times 10^5$	$F_4 = 0.26$

A computer-controlled stepper motor system using Labview, which runs under the Windows XP operating system, is employed to drive the towing carriage (Figure 2.6.3). The speed of the towing carriage was set manually before each run using a Labview software program. The motion is provided by a 200 step per revolution stepper motor. The stepper motor is driven with a micro-stepping IM804 stepper driver card and a PMK150S-24-U 24 Volts DC converter power supply. The driver is computer controlled using a National Instruments NI-DAQ 6024E acquisition card. The control computer is placed in a Pelican case fixed on the moving structure. A second computer is used to acquire data from the force transducer mounted on the sting. The velocity profile of the carriage speed can be seen in Figure 2.6.4.

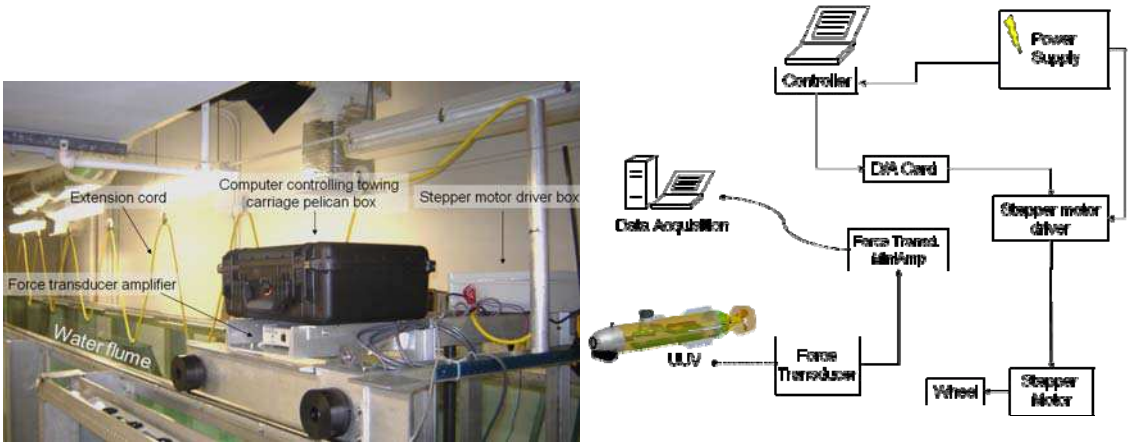


Figure 2.6.3: Experimental Arrangement: Towing Carriage, stepper motor controller and force transducer.

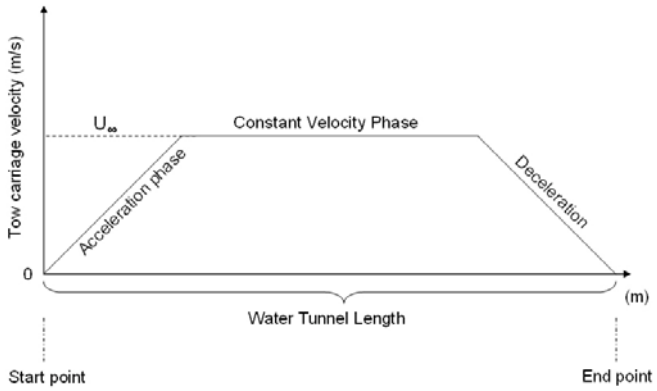


Figure 2.6.4: Towing carriage velocity profile.

The forces on the RPUUV were measured for a range of vehicle yaw ψ angles (Figure 2.6.5) and a rudder angle of $\delta = 0^o$ using a six component force transducer (AMTI UDW3-6-100). The RPUUV model was suspended from the towing carriage at a depth of 0.3 [m] (2.0 model diameters from both the free surface and bottom of the test section). The force transducer and sting were incorporated such that as the model is rotated in yaw, the x-y axes of the model were always parallel to the x-y axes of the force transducer (Figure 2.6.6); the z-axes of the force transducer and model were collinear and pass through the hull midship point. The sting is engineered to hold the RPUUV model centered between the bottom of the towing tank and the free surface of the water and to limit the interference created by the structure itself. An optical rotation stage mounted at the top of the sting permits ψ to be set with an accuracy of $\pm 0.5^o$.

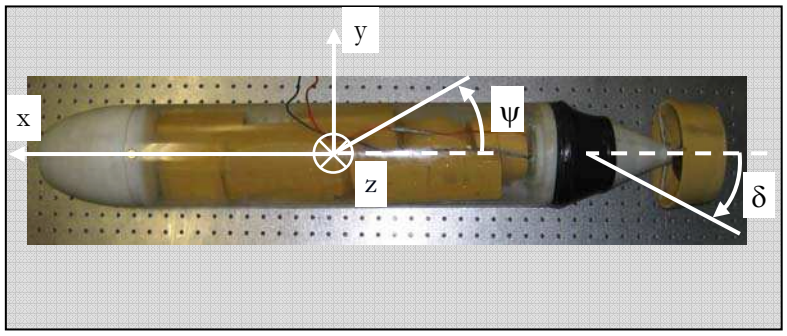


Figure 2.6.5: Definition of angles and coordinate system.

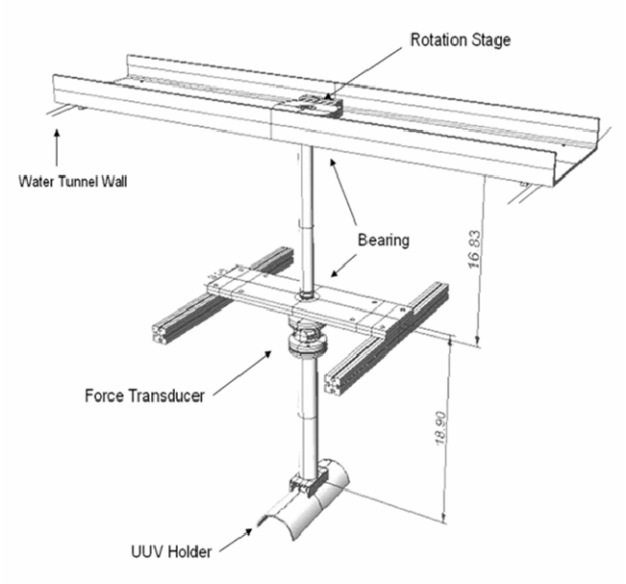


Figure 2.6.6: RPUUV sting design.

The hydrodynamic coefficients of lift, drag and yaw moment on the vehicle were measured for $\delta = 0^o$, $0^o \leq \psi \leq 30^o$. The effects of sting drag are removed from the force data by subtracting the average drag measured on the sting alone at $\psi = 0^o$. For all experiments, the data is processed and analyzed using Matlab.

2.6.4 Experimental Results

2.6.4-1 Hydrodynamic Coefficients

The drag force (i.e. total resultant force in the downstream direction), lift force (the resultant force perpendicular to the freestream direction) and yawing moment are measured as the yaw angle Ψ is varied at different freestream velocities. To obtain the drag and lift coefficients (C_d and C_l, respectively), the forces are normalized by the product of the dynamic pressure $0.5\rho U^2$, and the RPUUV frontal area $A_f = \pi d_v^2/4$. The yaw moment is normalized by $0.5\rho U^2 A_f l_v$ to find the coefficient C_m.

The value of C_d at $\Psi = 0^o$ can be predicted using the relation

$$C_D = C_f \left[1 + 60 \left(\frac{d_v}{l_v} \right)^3 + 0.0025 \left(\frac{l_v}{d_v} \right) \right],$$

for the drag coefficient C_D based on total wetted surface area [2].

Here C_f is the Reynolds-number-dependent turbulent skin friction coefficient for a flat plate, for example as obtained using the I.T.T.C. line [3]. After converting C_D to C_d (drag coefficient based on A_f), the theory predicts $C_d \approx 0.2$ at a Reynolds number R_2. The experimentally determined values of $C_d = 0.2802$ at $\Psi = 0^o$ compare favorably (Figure 2.6.7), especially at the higher Reynolds numbers.

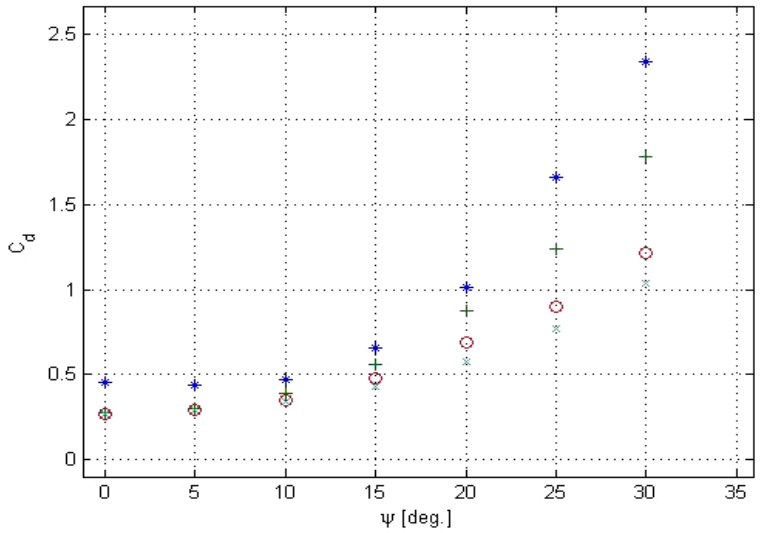


Figure 2.6.7: RPUUV drag coefficient as a function of yaw angle ψ measured at R_1 (*), R_2 (+), R_3 (o) and R_4 (x).

The duct around the propeller can be modeled as a 3D wing with a NACA 21016 hydrofoil profile (Figure 2.6.10) held by 3 cylinders (Figure 2.6.8).

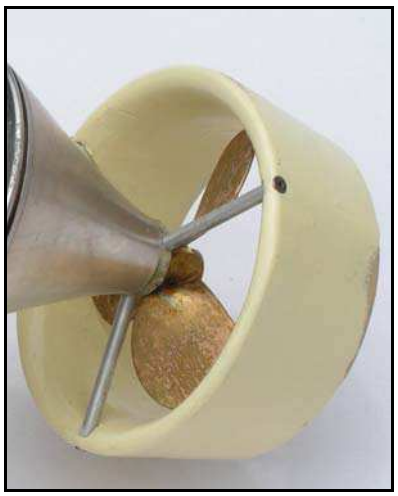


Figure 2.6.8: Close-Up Photograph of RPUUV Tail Section.

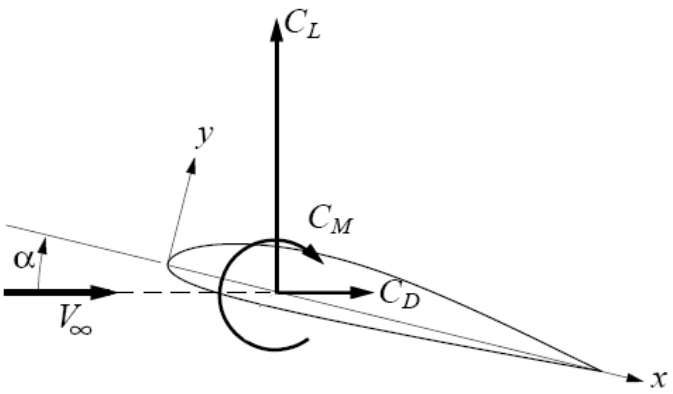


Figure 2.6.9: Force and moments applied on a hydrofoil.

The coefficient of drag of the wing profile normalized by the RPUUV dimensions is $C_{d\ 2Dwing\ profile} = 0.1027$ in two dimensions (Figure 2.6.10). The drag prediction for a ring fin is based on the effective aspect ratio (AR = 4D/(πc)) and the three dimensional lift coefficient. The effective area of the ring is taken as $A_e = \pi Dc/2$. The coefficient of lift C_l of a thin two dimensional airfoil is $C_l = C_l'$, where α is the angle of attack of the wing and C_l' is

$$C_l' = \frac{1}{0.63 + \frac{1}{\pi(AR)}}.$$

From classical wing theory

$$C_d = \frac{C_l^2}{\pi (AR)}.$$

In addition to the wing, the three cylinders holding the duct to the body of the RPUUV have to be taken in consideration. The theoretical drag coefficient is

$$C_d = \frac{D}{\frac{1}{2}\rho U^2 l} \times 3 \; cylinders.$$

The coefficient of drag at Reynolds number equal to R_2 of the duct normalized by the RPUUV dimensions is $C_{dw} = 0.089$, reducing the total coefficient of drag of the entire vehicle (without duct) to $C_d = 0.2703$. At R_1, the drag coefficient of the duct normalized by the RPUUV dimensions is $C_{dw} = 0.059$.

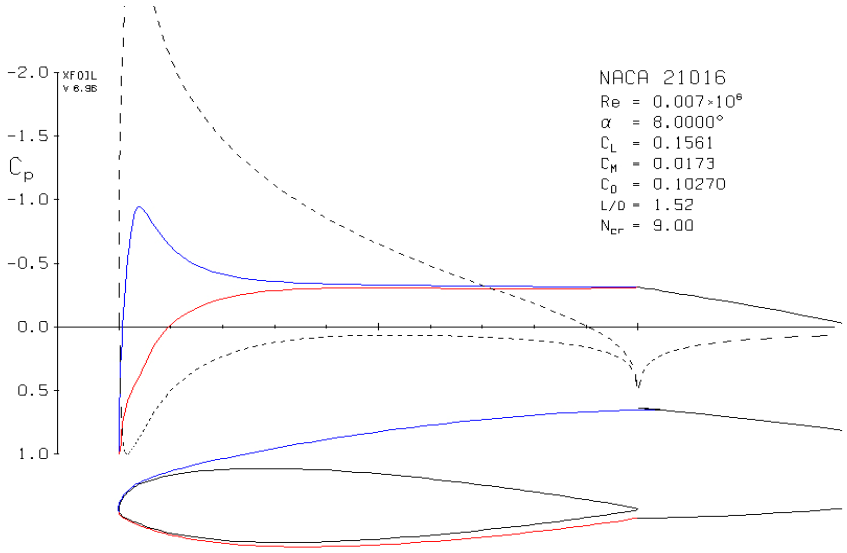


Figure 2.6.10: The NACA 21016 hydrofoil profile.

Low Reynolds number measurements

Experiments were performed using the model of the RPUUV with and without the duct to confirm the drag estimates. The drag coefficient measurements at Reynolds numbers R_1 and R_2 are shown in Figures 2.6.11 and 2.6.12.

When the yaw angle is $\Psi = 0°$ and the thruster angle is $\delta = 0°$ one can see that the drag coefficient drops substantially when the propeller duct is removed. At a Reynolds number of R_1, the experimental measurements agrees very well with theory ($C_{dw} = 0.059$).

Table 6.2.2: Propeller duct drag versus Reynolds number.

Reynolds Number	C_d (with duct)	C_d (without duct)	C_{dw}
R_1	0.498	0.280	0.051
R_2	0.359	0.447	0.079

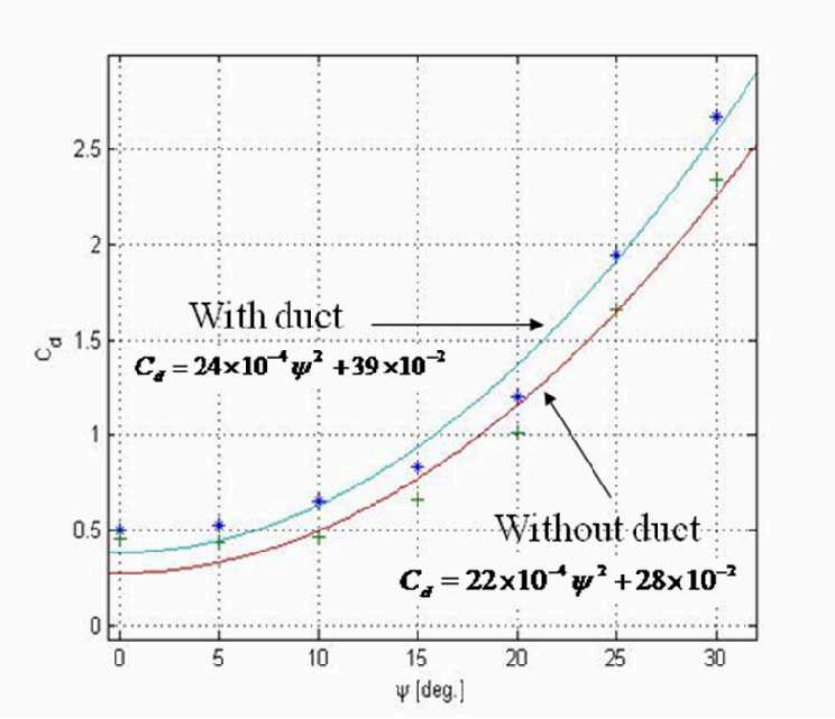


Figure 2.6.11: Drag coefficient of the UUV as a function of the yaw angle ψ at a Reynolds number R_1.

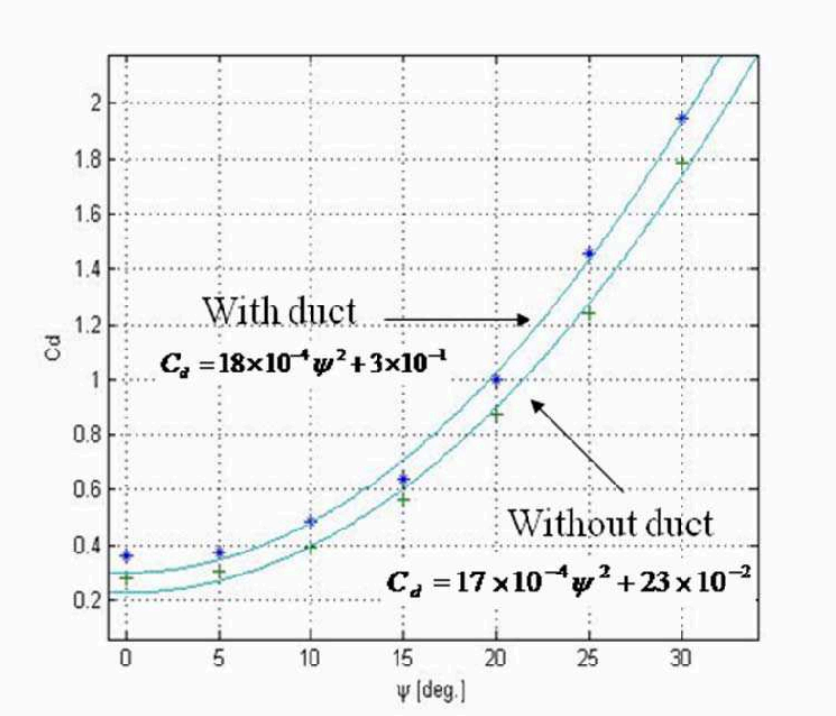


Figure 2.6.12: Drag coefficient of the UUV as a function of the yaw angle ψ at a Reynolds number R_2.

Without the propeller duct, the yaw-moment coefficient (C_m) stays approximately the same from R_1 to R_2 (Figures 2.6.13-2.6.14). When the propeller duct is mounted on the RPUUV, the coefficient of moment is lower at both Reynolds numbers R_1 and R_2 (Figure 2.6.17 & Figure 2.6.18).

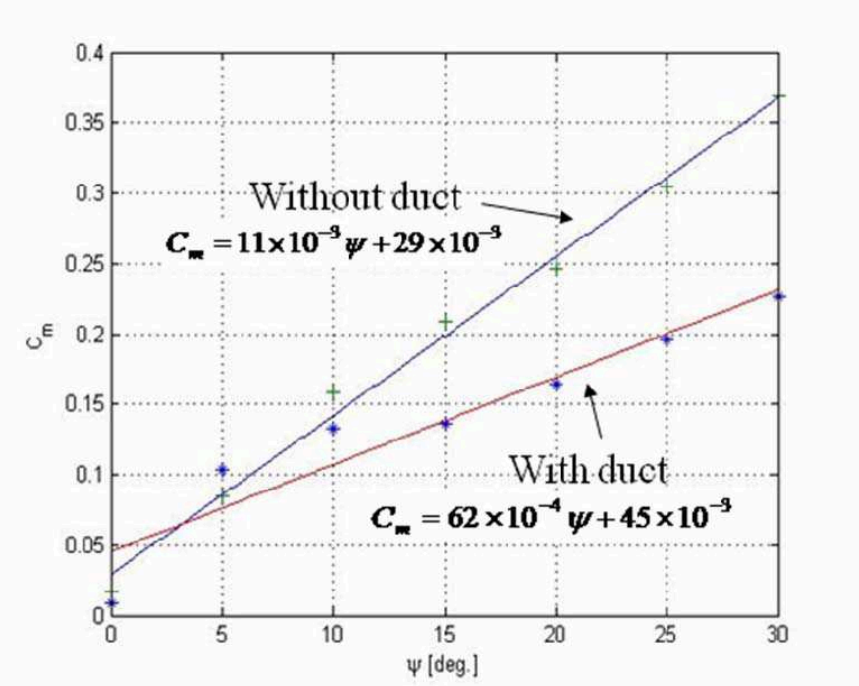


Figure 2.6.13: Yaw moment coefficient as a function of yaw angle ψ at R_1.

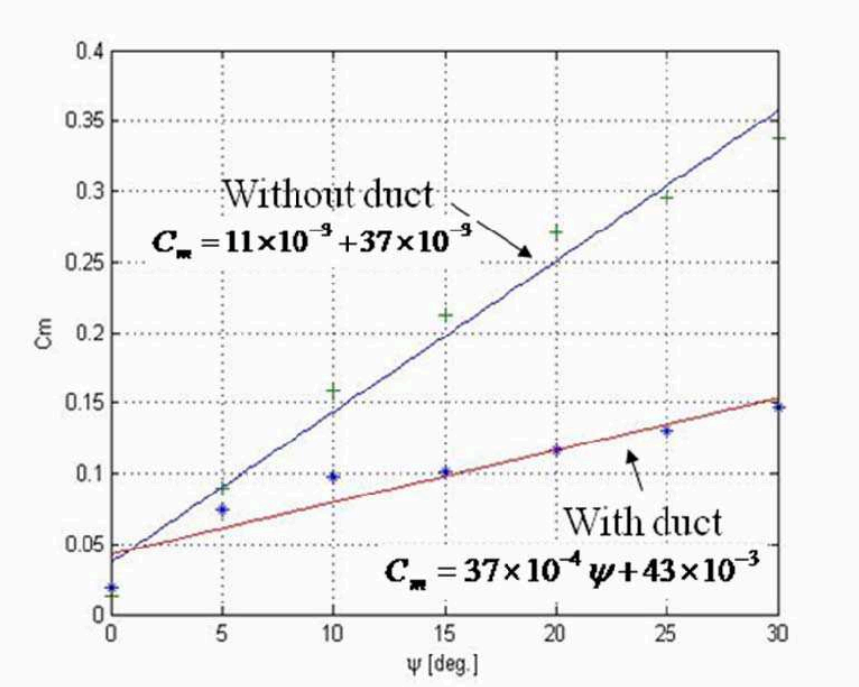


Figure 2.6.14: Yaw moment coefficient as a function of yaw angle ψ at R_2.

C_m is about 2 times higher without the duct than with it above a yaw angle of 5°. As shown in Figures 2.6.15 and 2.6.16, the difference of pressure around the body with or without the duct creates a yawing torque (known as a Munk moment) about the mid-section of the RPUUV. The

propeller duct creates a moment in the opposite direction (indicated by the red arrow), thus reducing the total yaw moment on the vehicle.

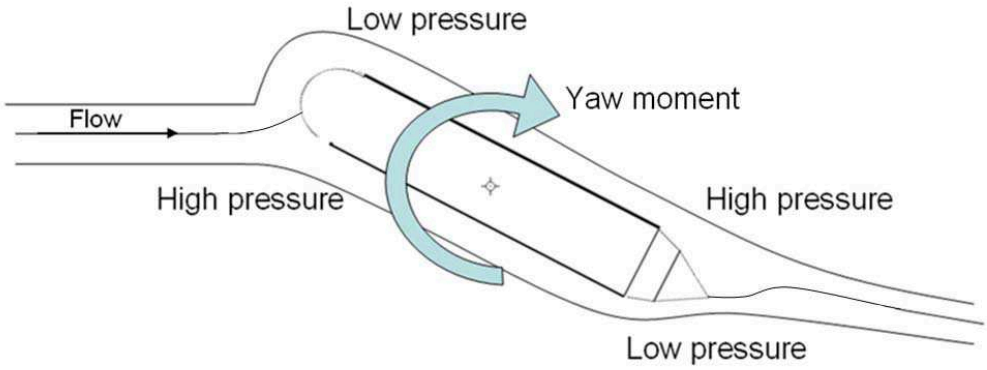


Figure 2.6.15: Pressure distribution along the body of the RPUUV without the duct.

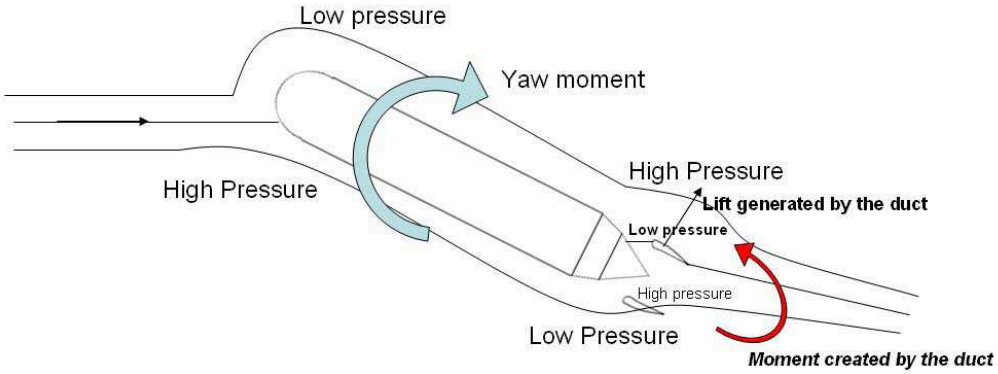


Figure 2.6.16: Pressure distribution along the body of the RPUUV with the duct.

The lift coefficient (C_l) increases from R_1 to R_2 at high yaw angles (above 15°)

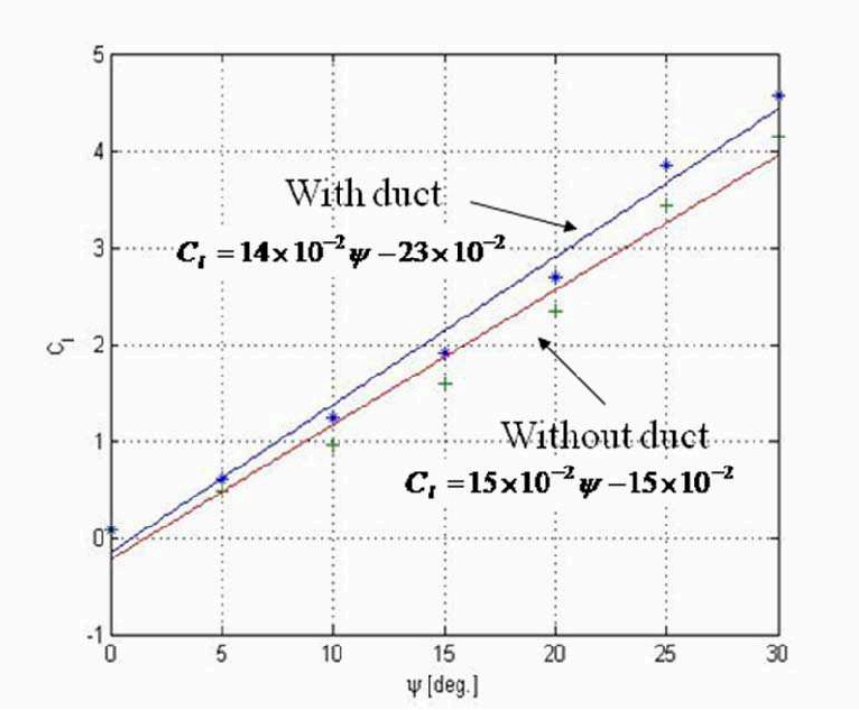


Figure 2.6.17: Lift coefficient as a function of yaw angle ψ at R_1.

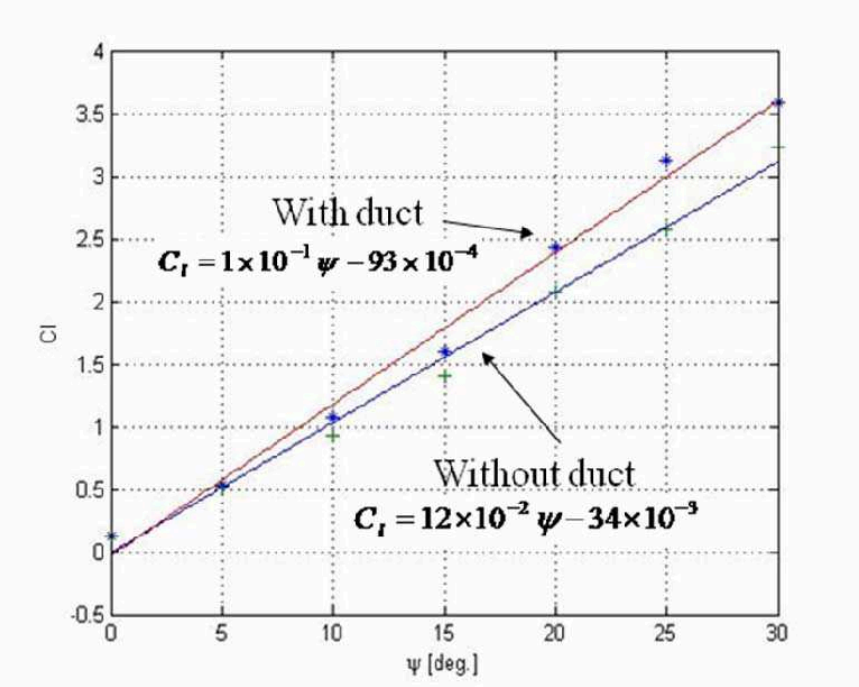


Figure 2.6.18: Lift coefficient as a function of yaw angle ψ at R_2.

High Reynolds number measurements

Experiments have been conducted at towing speeds of U_3 = 1 [knot] and U_4 = 1.5 [knots]. Following are the hydrodynamic coefficients measured during the tests (without propeller).

1 knot results

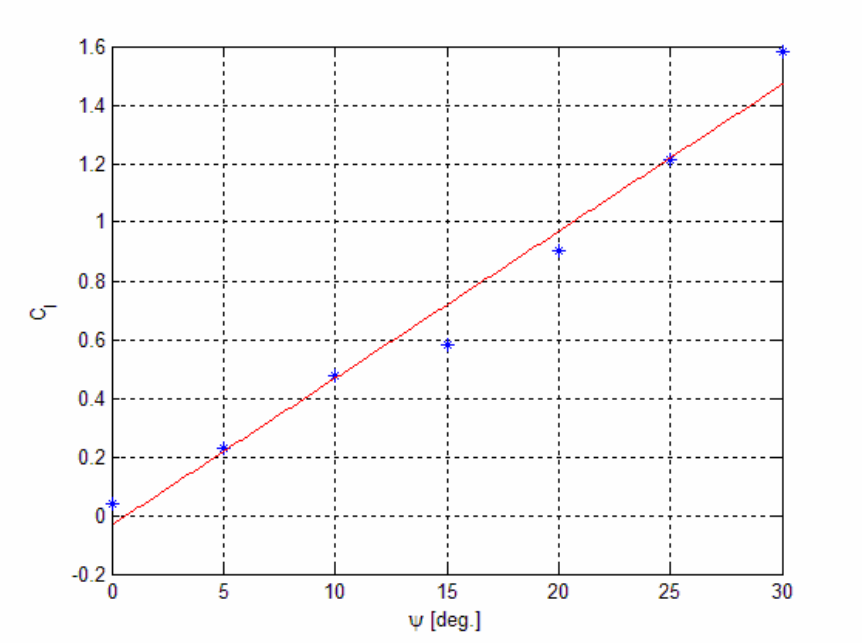


Figure 2.6.19: Lift coefficient as a function of yaw angle ψ at R_3 without duct $C_l = 5\times10^{-2}\,\psi - 34\times10^{-3}$.

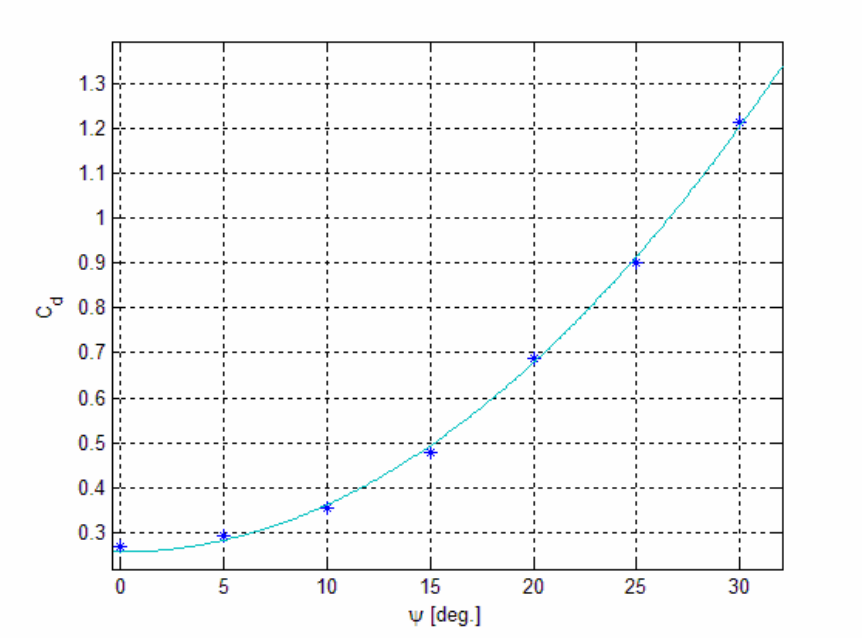


Figure 2.6.20: Drag coefficient as a function of yaw angle ψ at R_3 without duct $C_d = (11\times10^{-4})\psi^2 + 26\times10^{-2}$.

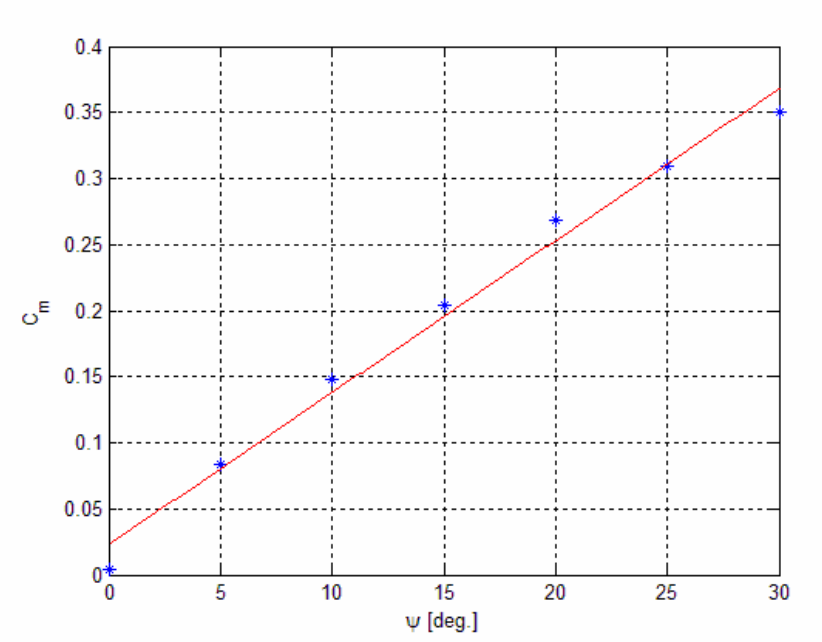


Figure 2.6.21: Yaw moment coefficient as a function of yaw angle ψ at R_3 without duct $C_m = 12\times10^{-3}\,\psi + 23\times10^{-3}$.

1.5 knot results

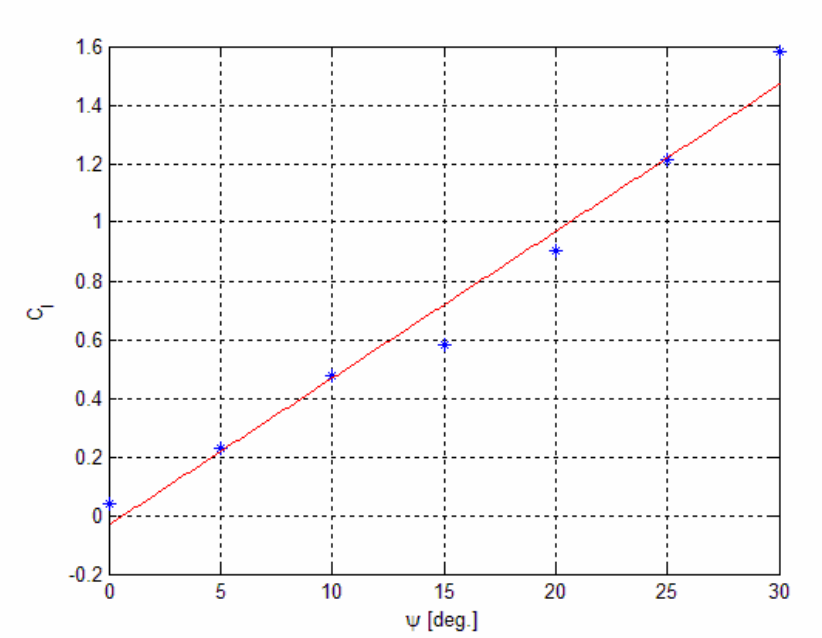


Figure 2.6.22: Lift coefficient as a function of yaw angle ψ at R_4 without duct

$$C_l = 49 \times 10^{-3}\, \psi - 28 \times 10^{-3}.$$

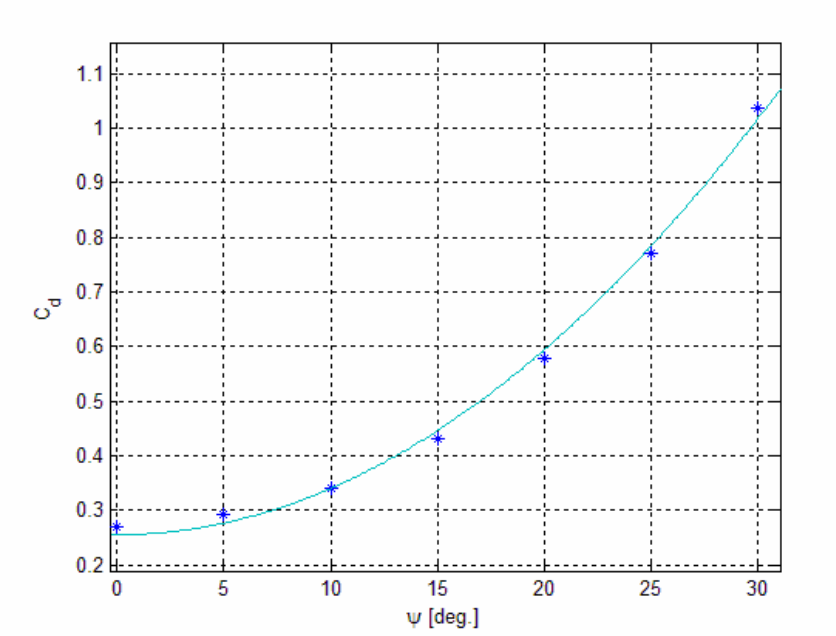


Figure 2.6.23: Drag coefficient as a function of yaw angle ψ at R_4 without duct

$$C_d = (85 \times 10^{-5})\, \psi^2 + 26 \times 10^{-2}.$$

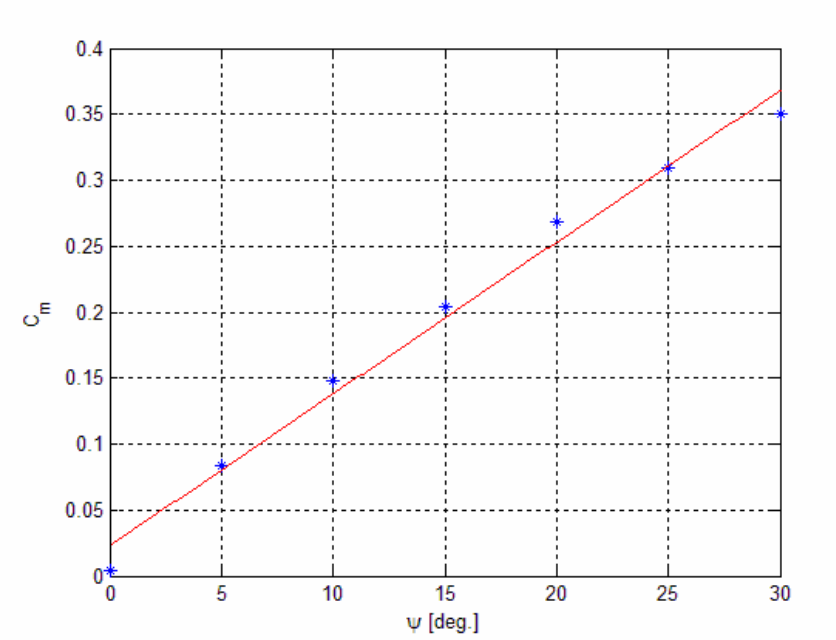


Figure 2.6.24: Yaw moment coefficient as a function of yaw angle ψ at R_4 without duct

$$C_m = 11 \times 10^{-3} \psi + 23 \times 10^{-3}.$$

As can be seen from Figures 2.6.26 and 2.6.29, the drag coefficient of the RPUUV at a Reynolds number of R_3 (C_d = 0.2692) and R_4 (C_d = 0.2693) is slightly lower than the one at a Reynolds number of R_2 (C_d = 0.359) (Figure 2.6.11). This difference appears to be due to the fact that there is a scattering of data in the transition range of Reynolds number between R = 10^5 and R = 2 x 10^5. In the transition from laminar to turbulent flow, the roughness of the body is a major factor that influences the transition between laminar and turbulent, since the frictional drag term is predominant in determining the drag of the vehicle [1].

2.6.5 Tests of Vehicle with New Sonar Side Panels

In year 3, an alternate configuration of the sonar transducers was proposed. The configuration includes two externally mounted transducer panels of dimension 17.8 x 55.9 x 2.5 [cm^3]. The centerline of each panel is parallel to the surge axis of the vehicle and is suspended at a distance of 5 [cm] from the outside of the RPUUV hull by a set of hinges that permit the panels to be rotated by an angle α from the vertical (Figure 2.6.25). The model RPUUV hull (without propeller and propeller duct attached) was outfitted with a pair of model sonar side panels (Figure 2.6.26) and tested at a speed of 1 knot for yaw angles of $0^o \leq \psi \leq 30^o$.

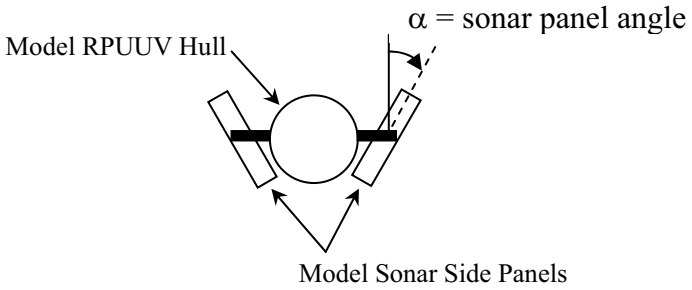


Figure 2.6.25: Schematic of RPUUV with sonar side panels – frontal view.

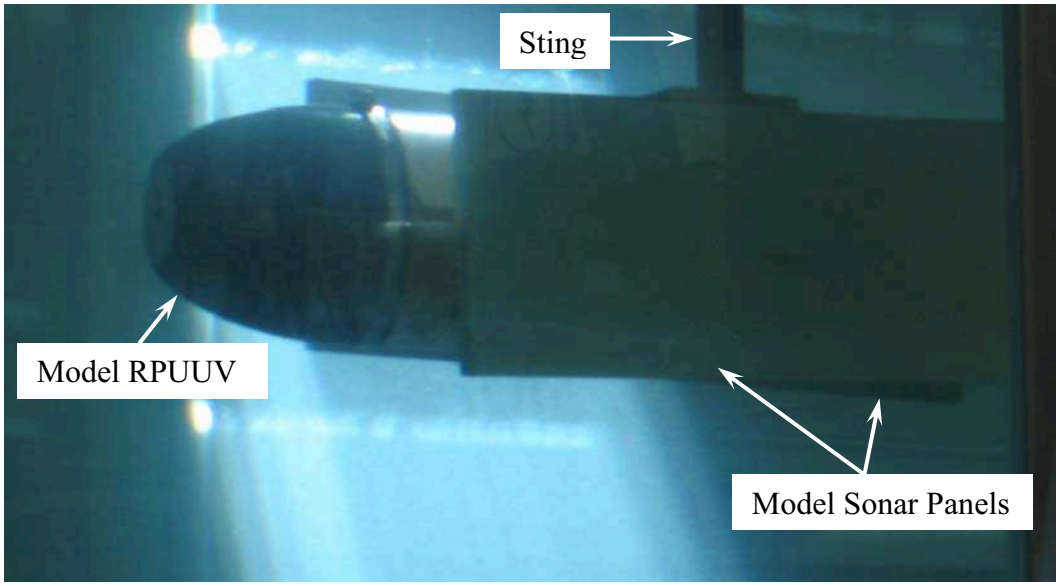


Figure 2.6.26: Modified model RPUUV hull with model sonar side panels mounted.

The resulting surge-, sway-force and yaw moment coefficients (C_d, C_{Fy} and C_m, respectively) are plotted in Figures 2.6.27 and 2.6.28 and listed in tables 2.6.2. By comparison with Figure 2.6.20, it can be seen that at $\psi = 0^o$ the panels substantially increase the drag on the vehicle. Additionally, the panels induce a large sway force that is expected to affect the controllability of the RPUUV. The values of surge force and moment coefficient are roughly the same at $\alpha = 0^o$ and $\alpha = 45^o$, however it can be seen that with the sonar panels mounted in the vertical position ($\alpha = 0^o$) the sway force is about twice as high as it is when the panels are canted at $\alpha = 45^o$.

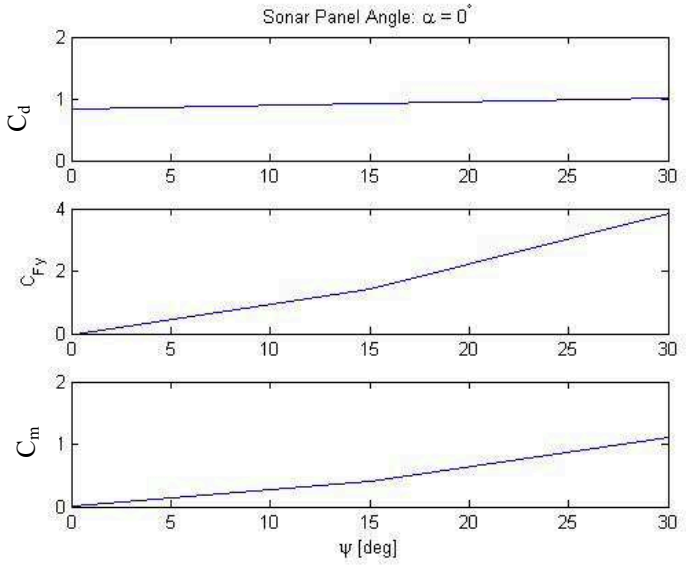


Figure 2.6.27: Force and moment coefficient variation with yaw angle ($\alpha = 0^o$).

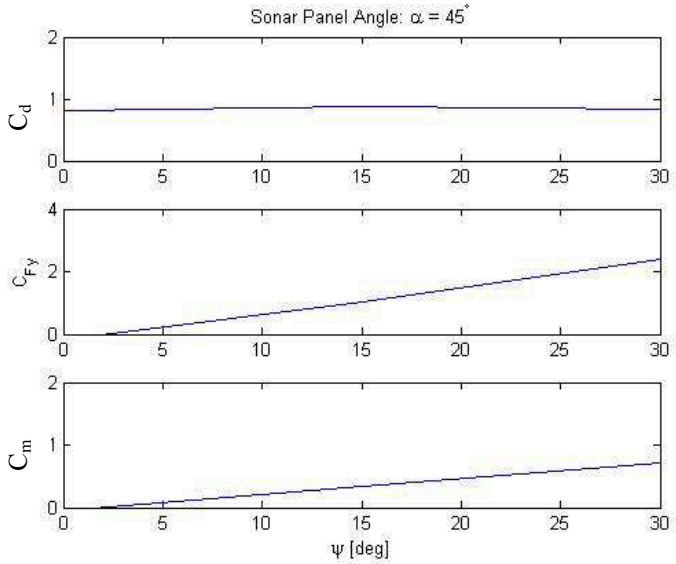


Figure 2.6.28: Force and moment coefficient variation with yaw angle ($\alpha = 45^o$)

Table 2.6.2: Force/Torque coefficients as a function of yaw angle ψ and panel angle α at a speed of 1 knot. C_d and C_m are the coefficients of drag and moment, respectively, as above. C_{Fy} is the force component along the sway axis.

Sonar Side Panels Vertical $\alpha = 0^o$

ψ [deg]	C_d	C_{Fy}	C_m
0	0.83	-0.04	0.02
15	0.91	1.43	0.41
30	1.01	3.85	1.12

Sonar Side Panels Angled $\alpha = 45^o$

ψ [deg]	C_d	C_{Fy}	C_m
0	0.80	-0.18	-0.05
15	0.88	1.05	0.35
30	0.83	2.40	0.72

2.6.6 Conclusions and Future Recommendations

A complete set of measurements of the hydrodynamic coefficients on the vehicle over the entire range of operating speeds has been performed. Further, the separate contributions to the forces and moments affecting the RPUUV from the main hull and the propeller duct have been determined. The results are expected to be useful for predicting the open loop response of the vehicle and to aid in the development of a closed loop controller. In addition, the hydrodynamic effects of a new sonar side panel configuration have been tested.

The specific tasks carried out during Year 3, and presented in the report, are:

[a] Task 3.18: Construction of experimental models and modifications of flow facility mounting supports.

[b] Task 3.19: Experimental determination of the hydrodynamic characteristics and dynamics coefficients of the model hull form in a 4'x4' towing tank.

[c] Task 3.20: Testing of control surface configurations (propeller duct) to optimize the design.

[d] Task 3.21: Experimental data, an evaluation of the hydrodynamic design and recommendations for future innovations are included in this year 3 final report.

Key findings and future design recommendations include:

- The drag of the propeller duct and its cylindrical supports represents about 16-20% of the total drag on the vehicle. This contribution can be reduced by using faired duct support rods, rather than one with cylindrical cross sections.
- In general, a critical issue for vectored thruster designs is that tight turning radii can cause the vehicle to tumble tail-over-nose, if the operator is not careful or if the thruster rudder and elevation angles are not rate-limited. The propeller duct mitigates this effect somewhat by producing a moment that counteracts the Munk moment on the RPUUV hull when in a turn. Thus, if maneuverability is found to be an issue, increasing the chord length of the propeller duct slightly could help to mitigate the vehicle's propensity to tumble.
- The new sonar panel design introduces some hydrodynamic complications. Especially of concern are the drag penalty (C_d is roughly doubled) that will reduce the vehicles range and the large sway force generated by the panels in a turn. If controllability under the sway force is found to be an issue, an increase in the propeller duct length may be required.

REFERENCES FOR SECTION 2.6

[1] Ackermann, L. E. J. *Thrust Response of a Vectored-Thruster Unmanned Underwater Vehicle.* M. S. Thesis, Florida Atlantic University, Boca Raton, FL USA 2007.

[2] Nahon, M. "A simplified dynamics model for underwater vehicles,*" IEEE Symp. Autonomous Underwater Vehicles Technology*, 1996, pp. 373-379.

[3] Newman, J. N. *Marine Hydrodynamics.* MIT Press, Cambridge, MA USA, p. 31, 1977.

[4] van Manen, J. D. and van Ossanen, P. *Principles of Naval Architecture: Vol II.* Ed. Lewis, E. V. SNAME, pp. 213-225, 1998.

2.7. Hydrodynamics and Dynamics Analyses of the Remotely-Piloted Unmanned Underwater Vehicle (RPUUV) PI: Dr. P. Ananthakrishnan

Task 3.22

2.7.1. Introduction

This section of the report presents the computational works carried out in the area of dynamics and hydrodynamics of the RPUUV (remotely piloted unmanned underwater vehicle) during Year 3 (2007-2008) of the project. During this period, efforts were focused on determining effects of surface waves on the dynamics of RPUUV and the possibility of adding pair acoustic-array panels, each of dimension 22" x 7" x 1", on the sides without too adversely affecting RPUUV vehicle motion.

2.7.1.1 Basic Vehicle Characteristics

In the present form, the RPUUV consists of cylindrical middle body, hemi-spherical nose section and a conical tail section. A vectored thruster is used for forward motion as well as for maneuvering in both horizontal and vertical motions. The vehicle has an acoustic modem for underwater communication. A sketch of the vehicle, illustrating the main features of the vehicle, is given in Figure 2.7.1.1.

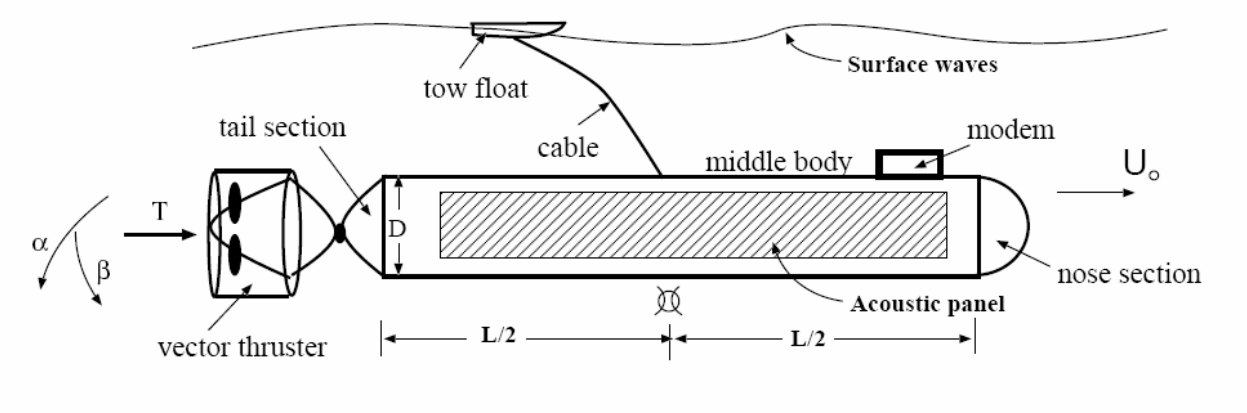


Figure 2.7.1.1 Illustrative sketch of the RPUUV

The middle body length, chosen as the characteristic length in hydrodynamic calculations is denoted as L and the vehicle diameter as D. Computations and simulations were carried out for various lengths and parameters. Results given in this report correspond to following parameter values:

- Middle-body length, L = 0.85 [m]
- Diameter, D = 0.16 L = 0.136 [m]

- Tail length = 0.13 L = 0.11 [m]
- Nose section length = 0.08 L = 0.068 [m]
- Water density, $\rho = 1025$ [kg/m^3]
- Acceleration of gravity, $g = 9.8$ [m/s^2]
- Vehicle volume= 0.0219 L^3= 0.01345 [m^3]
- Vehicle mass (for neutral buoyancy), m = 13.78 [kg] = 30 [lbf]
- Wetted surface area = 0.578 L^2= 0.42 [m^2]
- Projected area normal to x axis, A_p= 0.02 L^2= 0.0145 [m^2]
- Modem height = 2.5 inch = 0.063 [m]
- Modem diameter = 2.5 inch = 0.063 [m]

2.7.1.2 Vehicle Motion

As discussed in the Year 2 report, equations of rigid-body motion expressed with respect to body-fixed coordinates are integrated in time to simulate the RPUUV motion subject to thruster, fin and environmental forces. A Green's function based boundary-integral method is solved to determine the hydrodynamic coefficients and the added-mass forces and moments. The formulation is presented in Section 2.7.2

2.7.1.3 Wave Exciting Force

As the RPUUV is small compared to lengths of most prevailing ocean surface waves, one can neglect wave scattering by the RPUUV and determine the wave exciting force by simply integrating the dynamic pressure of only the incident waves; the force of incident wave is also referred to as the Froude-Krylov wave force [1], [2]. In determining RPUUV motion response, the wave radiation forces were neglected and infinite-fluid hydrodynamic coefficients used to estimate the RPUUV response to wave forces. The method of determining wave exciting force is presented in Section 2.7.3 of this report.

2.7.1.4 Motion Simulations

In Year 3, simulations of RPUUV motion both in horizontal and vertical planes were carried out to determine (i) effect of surface waves and (ii) possibility of adding flat-plate acoustic arrays on the sides without too adversely affecting vehicle performance. The simulations and findings are discussed in Section 2.7.4 of the report.

2.7.1.5 Contributions of the Project

Finally, in Section 2.7.5 of the report, the contributions of the present work to the design and improved performance of the RPUUV are summarized, including that made during Year 1 and 2.

2.7.2. Formulation of Vehicle Motion

For completeness, the equations of rigid-body motion formulated using body-fixed frame of reference are briefly reviewed in this section. The body-fixed coordinates oxyz is as shown in Figure 2.7.2.1 with the x axis, which is the axis of symmetry, being positive forward, the y axis positive from port to starboard and the z axis positive downward. The x, y and z components of translational velocity are denoted as u, v and w and the rotational velocity as p, q and r. The steady forward speed y is denoted as Uo. Propeller thrust is denoted as T.

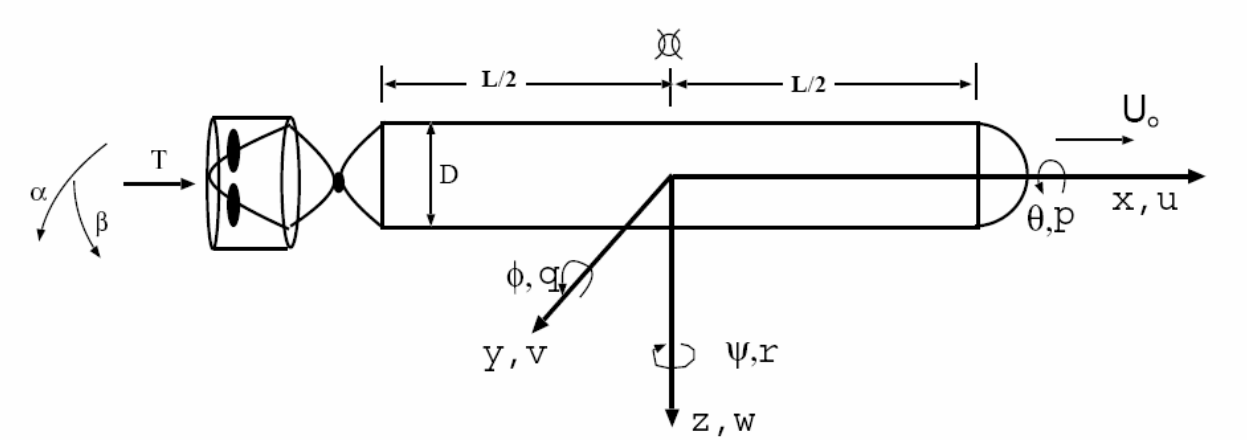


Figure 2.7.2.1 Body-fixed coordinates and notations used in the RPUUV dynamics formulation

2.7.2.1 Six DOF Rigid Body Equations of Motion

The 6DOF equations of rigid body motion with respect to body-fixed coordinates are given by [1], [3]

$$m[\dot{u} - vr + wq - x_G(q^2 + r^2) + y_G(pq - \dot{r}) + z_G(pr + \dot{q}) = \mathcal{X}$$

$$m[\dot{v} - wp + ur - y_G(r^2 + p^2) + z_G(qr - \dot{p}) + x_G(qp + \dot{r}) = \mathcal{Y}$$

$$m[\dot{w} - uq + vp - z_G(p^2 + q^2) + x_G(rp - \dot{q}) + y_G(rq + \dot{p}) = \mathcal{Z}$$

$$
\begin{aligned}
I_x\dot{p} + (I_z - I_y)qr - (\dot{r} + pq)I_{xz} &+ (r^2 - q^2)I_{yz} + (pr - \dot{q})I_{xy} \\
&+ m[y_G(\dot{w} - uq + vp) - z_G(\dot{v} - wp + ur)] = \mathcal{K}
\end{aligned}
$$

$$
\begin{aligned}
I_y\dot{q} + (I_x - I_z)rp - (\dot{p} + qr)I_{xy} &+ (p^2 - r^2)I_{zx} + (qp - \dot{r})I_{yz} \\
&+ m[z_G(\dot{u} - vr + wq) - x_G(\dot{w} - uq + vp)] = \mathcal{M}
\end{aligned}
$$

$$
\begin{aligned}
I_z\dot{r} + (I_y - I_x)pq - (\dot{q} + rp)I_{yz} &+ (q^2 - p^2)I_{xy} + (rq - \dot{p})I_{zx} \\
&+ m[x_G(\dot{v} - wp + ur) - y_G(\dot{u} - vr + wq)] = \mathcal{N}
\end{aligned}
$$

In the above equations m denotes the vehicle mass and (Ix, Iy and Iz) the mass moments of inertia about x, y and z axis, respectively. The coordinates of the center of gravity are denoted as (x_G, y_G, z_G). The x,y and z components of the resultant external force are denoted as *X, Y* and *Z*, respectively. The components of the moment of the external force about x, y and z axes are denoted as *K, M* and *N*, respectively. The over-dot represents time derivative.

2.7.2.2 Horizontal Plane Three DOF Equations of Motion

The primary modes affecting the motion on the horizontal plane are surge, sway and yaw. Setting other motions to be zero, we can obtain the following equations for the horizontal plane motion:

$$
\begin{aligned}
&m(\dot{u} - vr - x_G r^2 - y_G\dot{r}) = \mathcal{X}, \\
&m(\dot{v} + ur + x_G\dot{r} - y_G r^2) = \mathcal{Y}, \\
&I_z\dot{r} + m[x_G(\dot{v} + ur) - y_G(\dot{u} - vr)] = \mathcal{N}
\end{aligned}
$$

In the case of the plane motion, note that r = d ψ/dt where ψ denotes the Euler angle of displacement about the z axis (ie. heading angle).

2.7.2.3 Vertical Plane Three DOF Equations of Motion

The equations governing surge, heave and pitch motions are given by

$$
\begin{aligned}
m(\dot{u} + wq - x_G q^2 + z_G \dot{q}) &= \mathcal{X}, \\
m(\dot{w} - uq - z_G q^2 - x_G \dot{q}) &= \mathcal{Z}, \\
I_y \dot{q} - m[x_G(\dot{w} - uq) - z_G(\dot{u} + wq)] &= \mathcal{M}
\end{aligned}
$$

Note that the righting moment associated with the meta-centric height is included *M.*

2.7.2.4 Method of Analysis

The modeling of external force and moment, which includes forces from waves, is explained in Section 2.7.3 of the report. With initial values specified, the equations governing the body motion subject to external force and moment are time-integrated using the Euler's scheme to advance the solution in time and thereby simulate vehicle motion [3]. Upon determining velocity components in the body-fixed frame, the velocity components in the earth-fixed frame are obtained by appropriate coordinate transformation. The earth-fixed velocity components are then integrated with respect to time to determine vehicle trajectory and orientation as seen from earth-fixed frame of reference. The RPUUV vehicle motions are thus simulated.

2.7.3. Determination of Hydrodynamic Forces and Moments

In this section we describe methods used to determine wave forces, force on acoustic array flat panels and modeling of other hydrodynamic forces on the RPUUV.

2.7.3.1 Wave Force

Neglecting wave scattering, which is justifiable for small vehicles, the wave exciting force is determined by integrating the dynamic pressure of incident waves. The hydrodynamic force of incident wave is also referred to as the Froude-Krylov force [1] [2].

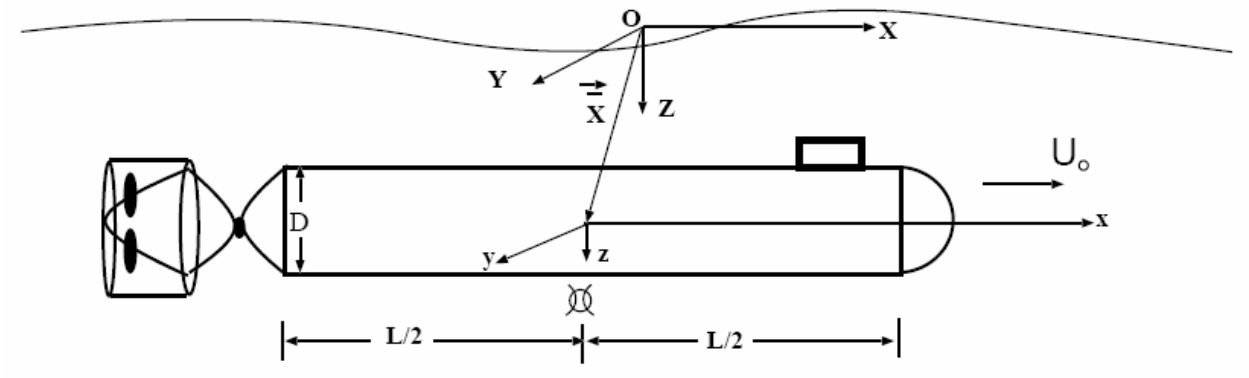


Figure 2.7.3.1 Body-fixed and earth-fixed coordinates for determining wave forces.

Consider a wave of height H, wave number k, frequency σ propagating along the positive X direction. The dynamic pressure of the wave is given by

$$p = \rho g \frac{H}{2}\, e^{-kZ} \sin(kX - \sigma t)$$

where ρ denotes water density, g the acceleration of gravity and Z the earth-fixed coordinate axis along the direction of g (see Figure 2.7.3.1). The Froude-Krylov force is then determined by surface integrating the dynamic pressure:

$$\vec{F} = -\int_S p\hat{n}\, dS$$

where S denotes vehicle surface and n the outward normal (opposite of pressure). Using the Gauss theorem, above integral can be written as a volume integral:

$$\vec{F} = -\int_\forall \nabla p\, d\forall$$

For small enough vehicle, the pressure gradient term can be approximated by that at the center of the vehicle. The integral then becomes

$$\vec{F} = \nabla\, \bar{p} \forall$$

where the over bar on p denotes that it is wave pressure gradient at the center of the vehicle (had the vehicle not been present!). Using the expression for pressure given above, one can obtain the following expressions for the wave-exciting Froude-Krylov force components:

$$F_X = \rho g k \frac{H}{2}\, e^{-k\bar{Z}} \cos(k\bar{X} - \sigma t)$$

$$F_Z = -\rho g k \frac{H}{2}\, e^{-k\bar{Z}} \sin(k\bar{X} - \sigma t)$$

where $(\bar{X}, \bar{Y}, \bar{Z}) = \vec{\bar{X}}$ denote the instantaneous earth-fixed coordinates of the center of the vehicle.

Components of the Wave Force in Vehicle-Fixed Coordinates.

We integrate the equations of motion, expressed in body-fixed coordinates, to simulate the vehicle motion. The components of the wave force in body-fixed coordinates are given by

$$F_x = F_X \cos\theta - F_Z \sin\theta$$

$$F_z = F_X \sin\theta + F_Z \cos\theta$$

where θ denotes to pitch angular displacement.

2.7.3.2 Hydrodynamic Force on Acoustic-Array Panels

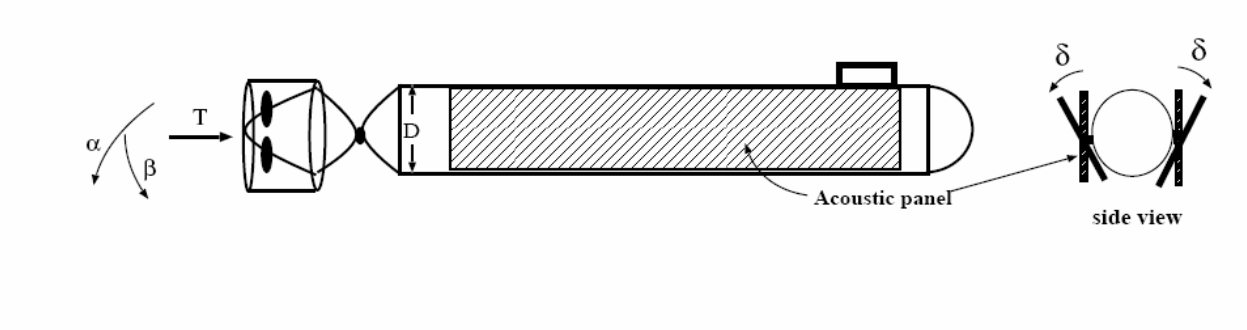


Figure 2.7.3.2 RPUUV with side panel acoustic arrays.

It is expected that the RPUUV will carry an acoustic array on flat panels and mounted to the side of the vehicle, as shown in Figure 2.7.3.2. Each panel is of dimension 22 in x 7 in x 1 in and can be tilted from vertical position from $\delta = 0^o$ to $d = 45^o$. In this section we present modeling of the hydrodynamic force on the panels.

We model the acoustic panels as lifting flat plates [2]. The lift force can be written as

$$L = \frac{1}{2}\rho C_L U^2 A_p$$

where the lift coefficient C_L of a flat plate is given by

$$C_L = 2\pi \sin\,\alpha \approx 2\pi\alpha, \text{ for small angle of attack, } \alpha.$$

In the above equation, A_p denote the area of the flat panel projected on the horizontal plane

$$A_p = 2.L.W.\sin\delta$$

where L denotes the length of the panel, W the width of the panel and δ the deflection of the panel from the vertical plane. The factor 2 is to account for the two panels, one on port side and the other on the starboard side of the vehicle.

For vehicle in vertical plane motion (surge, heave and pitch), the angle of attack of the panel is given by

$$\alpha = \tan^{-1}\,\frac{w}{u}$$

where u denotes the surge velocity and w the heave velocity of the vehicle. The resultant velocity magnitude U (appearing in the expression for lift force) is the amplitude of the vector sum of u and w. The lift force acting on the acoustic array panels is decomposed along vehicle's x and z directions to determine surge and heave components of the force on the RPUUV:

$$F_x = L\,\sin\alpha; \qquad F_z = -L\,\cos\alpha$$

2.7.3.3 Modeling of Other Forces.

Modeling and determination of other hydrodynamic forces, such as added-mass force, force on acoustic modem etc., were presented in Year 1 and 2 final reports. The principal vehicle and hydrodynamics related quantities used in the simulations presented in this report are summarized in the following table.

Item	Value	Remarks
Vehicle mass, m	13.78 [kg]	neutrally buoyant
Drag force at $U_O = 1$ [m/s]	1.5 [N]	$C_D = 0.2$ D = 1.5 U^2
Modem drag at $U_O = 1$ [m/s]	2.07 [N]	$C_D = 1.0$ D = 2.07 U^2
Modem drag moment at $U_O = 1$ [m/s]	0.207 [N m]	$C_D = 1.0$ M = 0.207 U^2
Mast drag at $U_O = 1$ [m/s]	2.44 [N]	$C_D = 1.0$ D = 2.44 U^2
Mast drag moment at $U_O = 1$ [m/s]	2.205 [N m]	$C_D = 1.0$ M = 2.205 U^2
Added mass μ_{11}	0.48 [kg]	in infinite fluid
Added mass $\mu_{22} = \mu_{33}$	11.6 [kg]	in infinite fluid
Added mass $\mu_{26} = \mu_{53}$	0.02 [kg-m]	in infinite fluid
Added mass $\mu_{55} = \mu_{66}$	0.62 [kg-m^2]	in infinite fluid

Note, the mast is no longer used on RPUUV vehicle as its presence was found to make the vehicle unstable and uncontrollable.

2.7.4. Simulation of Vehicle Motions: Discussions and Findings

In this section of the report, we present and discuss results of simulations of RPUUV motion carried out during Year 3 of the project. Even though simulations were carried out for a wide range of varying parameters, only representative and key results, which led to the findings reported, are presented in the report. All the simulations were carried out from t = 0 to t = 100 [s]. The thruster was ramp started from t = 0 [s] to reach specified thrust at t = 30 [s]. A time step size of $\delta t = 0.1$ [s] was used in the simulations.

2.7.4.1 RPUUV Motion in Waves

First, we present results for vertical plane motion (surge, heave and pitch) of the RPUUV under waves. Figure 2.7.4.1 shows vehicle trajectories in deep water waves all having wave number of k = 0.314 (ie., wave length = 20 [m]) and vehicle in head seas. The wave heights range from H = 0 (ie. no waves) to H = 0.5 [m] and the initial depth of vehicle submergence is 3 [m]. The vector thrust was set at T = 5 [N] and thrust angle β = 2 deg. In the absence of waves and without any control of vehicle motion, the vehicle covers a distance of 120 [m] horizontally and about 15 [m] vertically in a time span of 100 [s]. In the presence of waves, in particular when wave height is 0.5 [m], the vehicle undergoes an oscillatory motion when near the surface because of the effect of surface waves. When at sufficiently large depth (ie., greater than one half the wave length), the wave effect on vehicle motion becomes negligibly small.

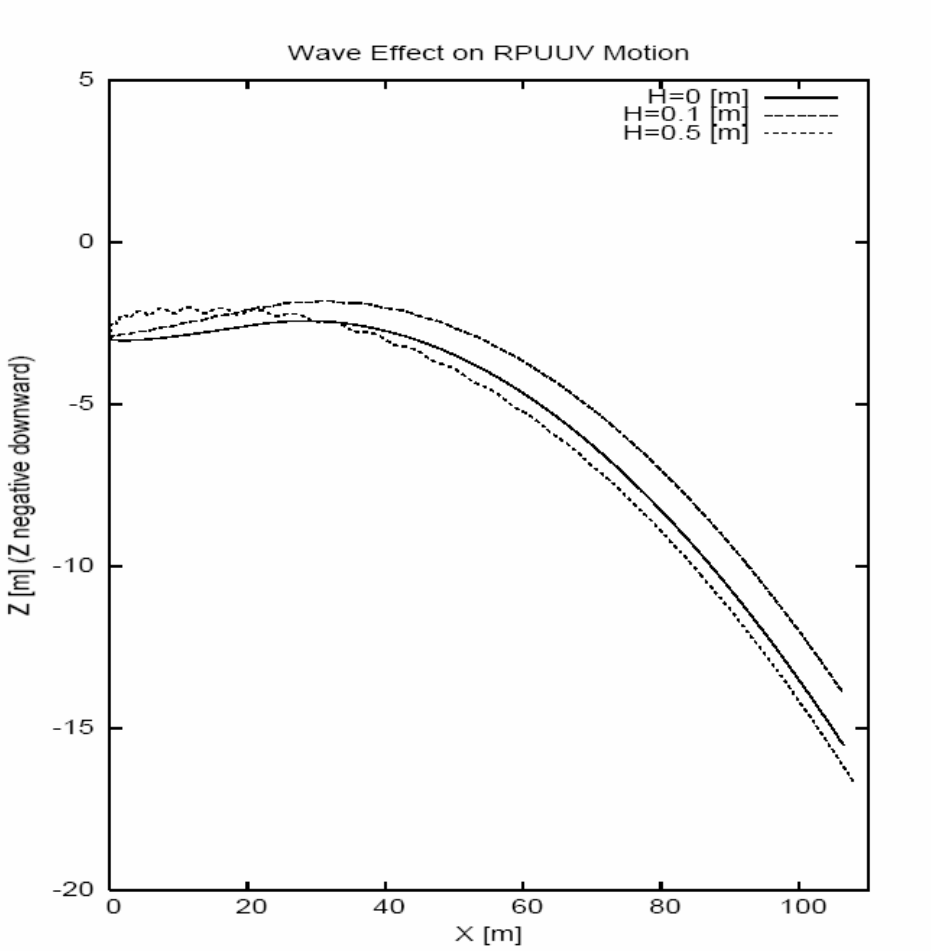


Figure 2.7.4.1 Trajectory of RPUUV Trajectory of RPUUV in waves of length L = 20 [m], heights = 0., 0.1, and 0.5 [m], initial depth of submergence 3 [m], vector thrust T = 5 [N] and vector thrust angle 2 [deg]

Time history of the wave force components corresponding to T=5 [N] (head seas), wave length = 20 [m], initial depth of submergence 3 [m] and wave height H = 0.5 [m] is given in Figure 2.7.4.2. The amplitude of the force decays rapidly as the vehicle moves downwards away from the free surface.

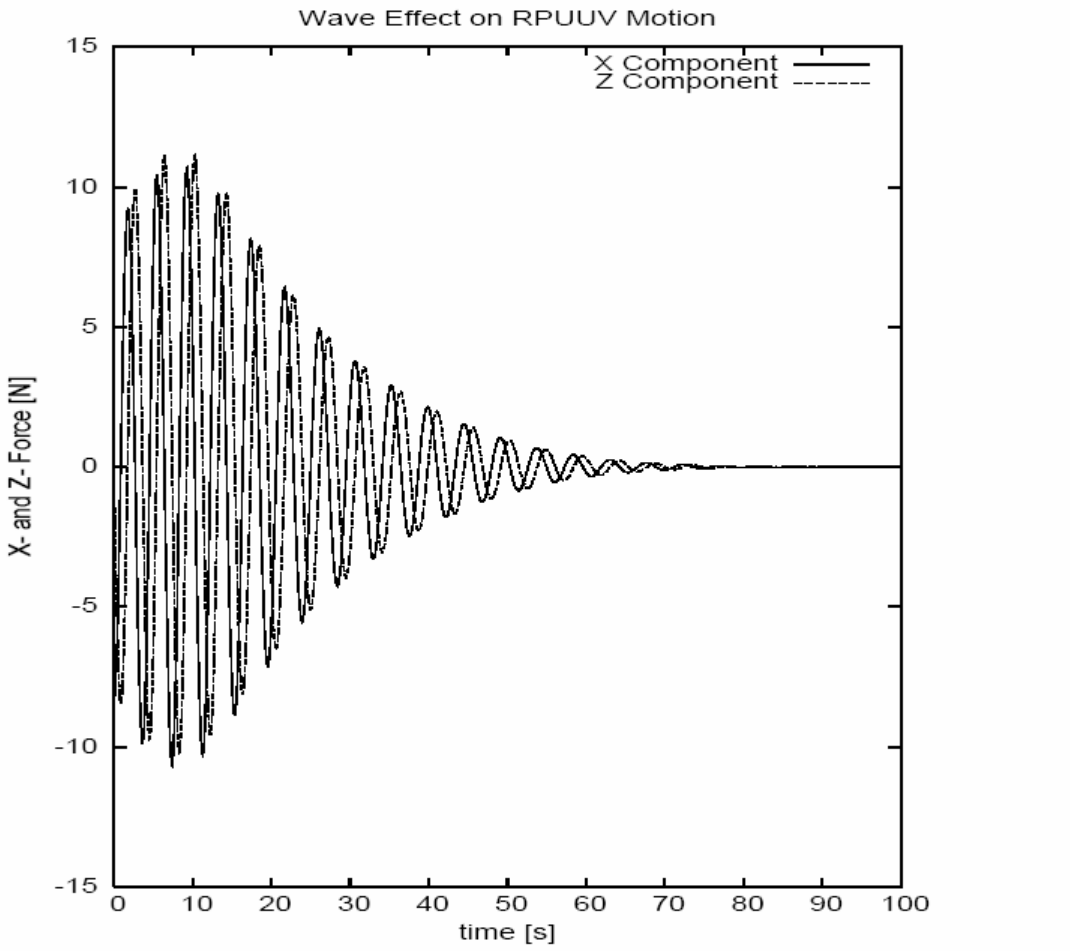


Figure 2.7.4.2 Time history of the X- and Z- components of the wave force: wave length L = 20 [m], wave height H = 0.5 [m] and initial vehicle submergence 3 [m].

Results corresponding to vehicle under thrust T = 5 [N] at thrust angle of 2 [deg] initially at epth of 1 [m] below the surface excited by deep water wave of length L = 10 [m] and height H = 0.1 is given in Figures 2.7.4.3 and 2.7.4.4. The vehicle initially rises to the surface before making the descent. The vehicle trajectory is oscillatory when near the surface. The time history of the wave force shows that the amplitude of the force increases initially before decaying, which is because of the initial approach of the vehicle towards the free surface.

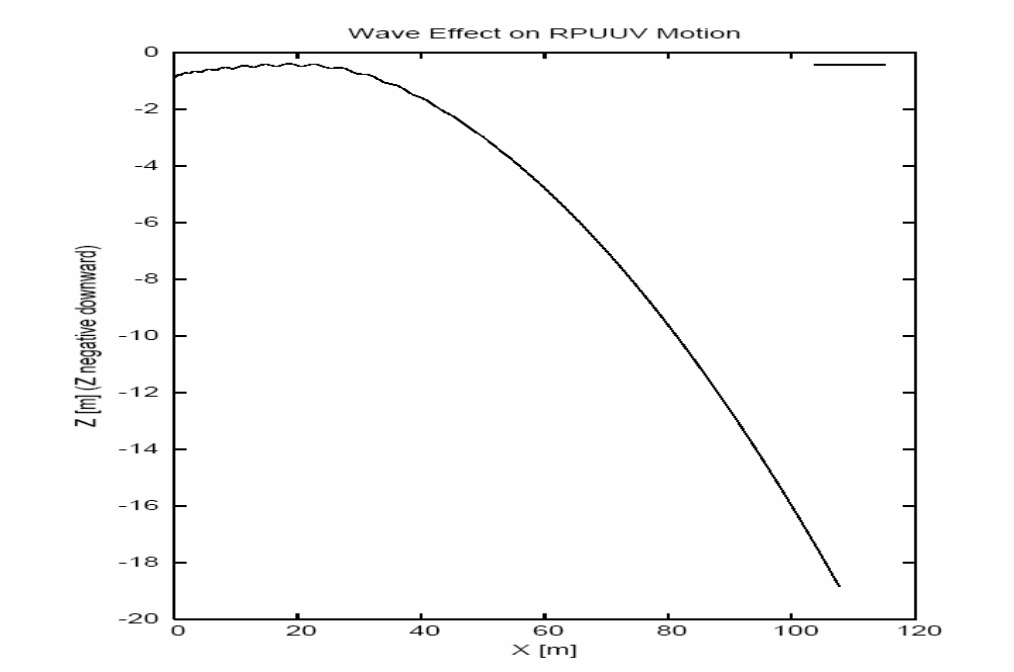


Figure 2.7.4.3 Trajectory of RPUUV in deep water wave of length L = 10 [m], height = 0.1 [m], initial depth of submergence 1 [m], vector thrust T = 5 [N] and vector thrust angle 2 [deg]

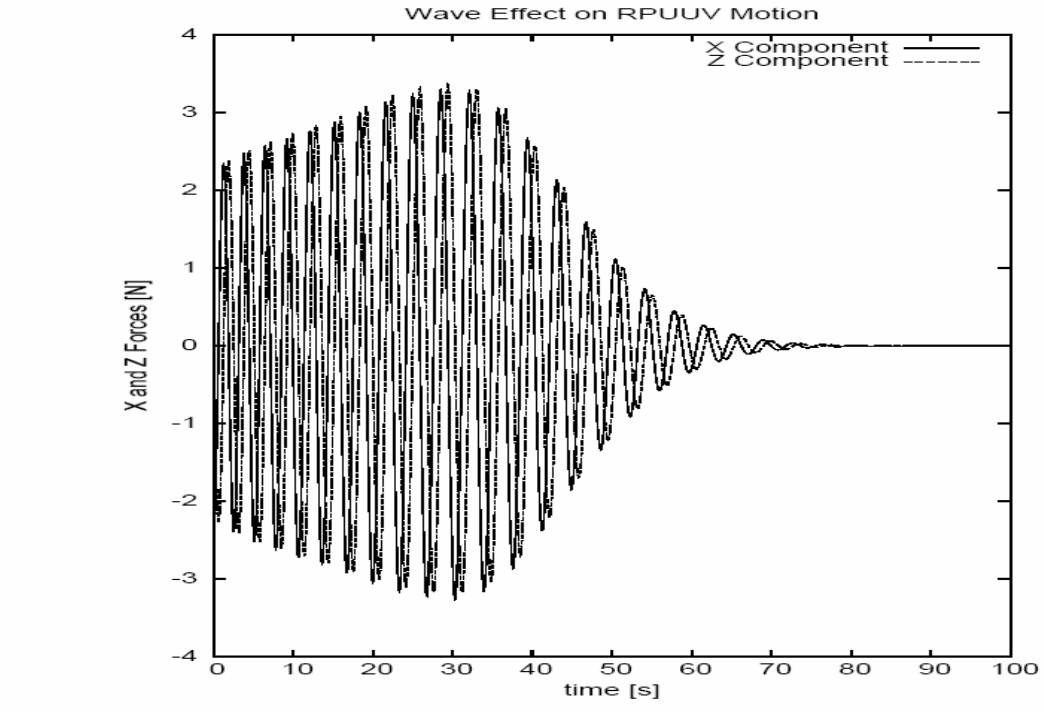


Figure 2.7.4.4 Time history of the X- and Z- components of the wave force: wave length L = 10 [m], wave height H = 0.1 [m] and initial vehicle submergence 1 [m]

Findings:

Based on the simulations of the vehicle motion under surface waves, we find that

- The RPUUV fitted with vectored thruster is maneuverable under surface waves.
- Once far below the surface (beneath a depth that is larger than the wave length), the effect of waves become negligible.
- Maintaining a horizontal motion very near the free surface will require automatic or manual closed-loop control of the vehicle.

2.7.4.2 Effect of Acoustic Sonar Array on RPUUV Motion

Next, the motion of the RPUUV in the vertical plane is simulated by including the proposed acoustic side panel arrays. Simulations are carried out for a range of parameters and the results highlighting the findings are presented.

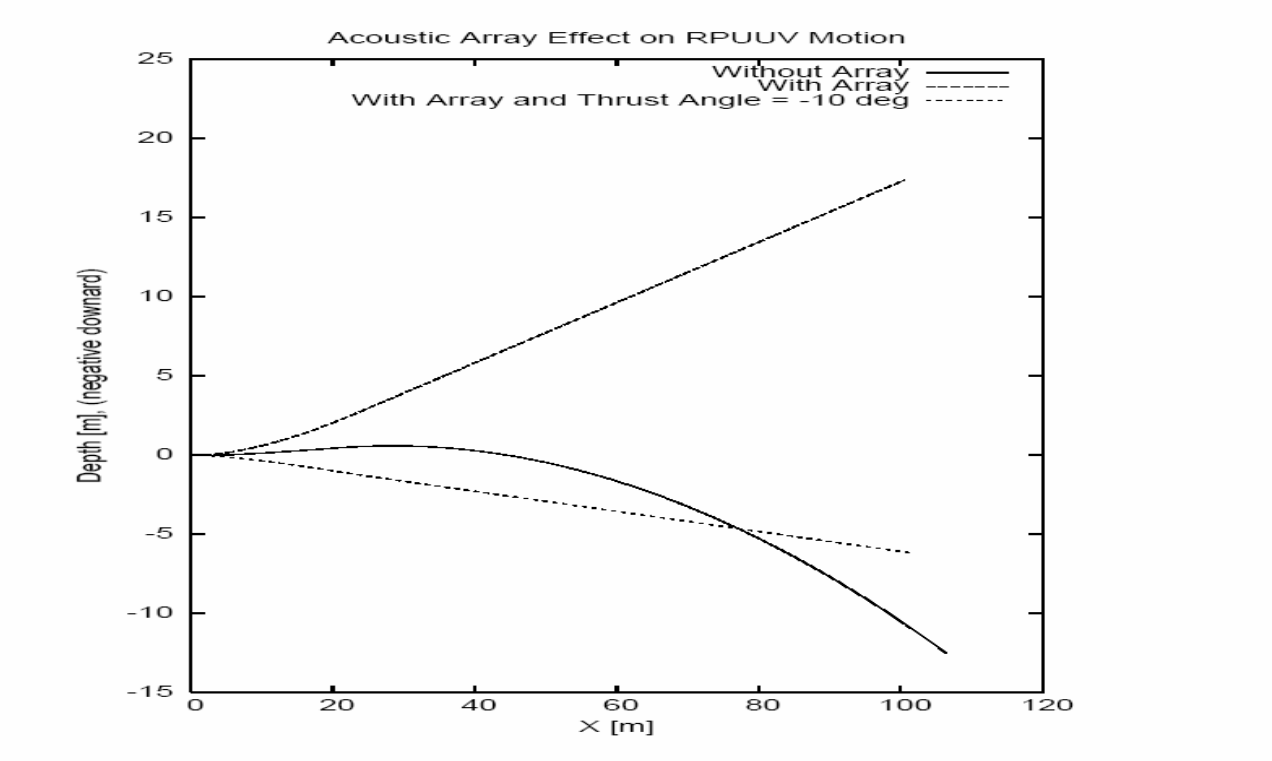


Figure 2.7.4.5 Trajectory of the RPUUV with and without side acoustic arrays: (i) without acoustic array, T = 5 [N]; (ii) with acoustic array at 10 [deg] inclination from the vertical; and (iii) with acoustic array at 10 [deg] inclination from the vertical and thrust angle = -10 [deg]

In Figure 2.7.4.5, results corresponding to cases with and without arrays are presented. Without acoustic array and with vector thrust angle of 2 [deg], the vehicle covers a horizontal distance of 100 [m] and a vertical downward depth of about 12 [m] during the 100 [s] of simulation. With the acoustic arrays included and inclined at 10 [deg] from vertical, the RPUUV undergoes an upward motion covering a height of about 18 [m]. But when the vector thruster angle is set at 10 [deg], the vehicle returns to the trajectory that is closer to the one without arrays. The result demonstrates that the force and moment induced by the acoustic arrays can be effectively countered by the vector thruster.

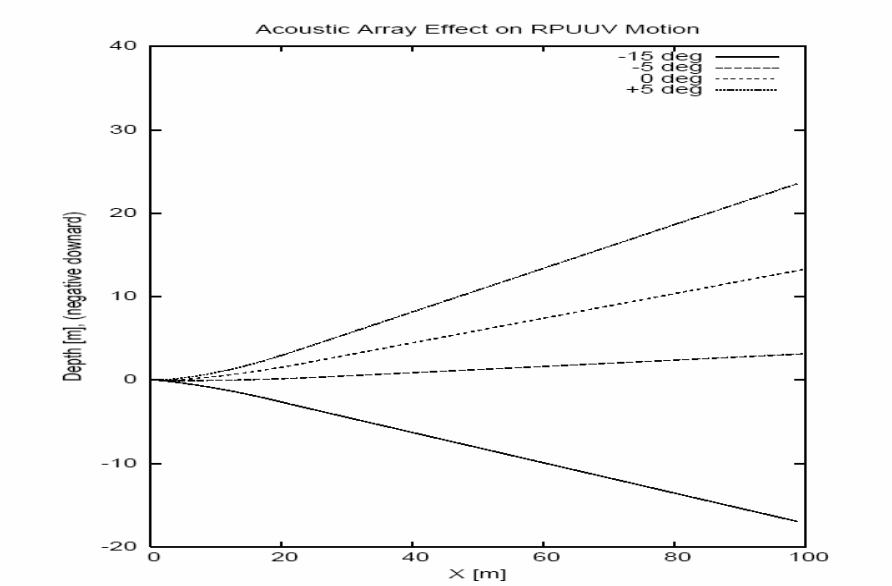


Figure 2.7.4.6 Trajectory of the RPUUV with side acoustic arrays with vector thrust T = 5 [N], inclination of acoustic panel = 30 deg from the vertical and for various angles of vector thruster (0, -15, -5, +5 deg).

The fact that vector thruster can effectively manage the forces induced by the acoustic panels is further demonstrated in the results presented in Figure 2.7.4.6. Here, the panels are inclined at 30 [deg] from the vertical. The thrust is set at 5 [N] for a range of thruster angles spanning from -15 [deg] to + 5[deg]. One can observe that with the help of the thruster the vehicle can be steered effectively even with inclusion of the acoustic panels.

Findings:

Based on the simulations including acoustic panels we find that

- Inclusion of the acoustic panels affects vehicle motion
- The effect of the acoustic panels on the motion can however be countered with the use of vectored thruster.

2.7.5 Conclusion

As discussed in this report, we have carried out the following investigations during Year 3 of the project

- Determination wave forces on the RPUUV
- Determination of RPUUV response to waves
- Simulation of RPUUV motion including side acoustic panels

The study resulted in the following findings and contributions to design and performance enhancement:

- The vehicle is maneuverable under waves.
- To maintain the vehicle on a horizontal plane while in close proximity to the free surface will require automatic or manual control of the vehicle.
- Vector thruster is quite effective for the maneuverability of the RPUUV even with side acoustic panels.

Some of the key tasks and contributions of the present study over the past three years include

- Determination of the possibility of maneuvering the RPUUV with a vector thruster.
- Determination of drag and lift forces on the RPUUV and its appendages.
- Determination of added-mass coefficients including sea bottom effects.
- Finding that the vehicle is maneuverable close to the sea bottom.
- Finding that the vehicle is unstable in the presence of the mast (designed to identify the location of the vehicle).
- Finding that the vehicle with side acoustic panels (each of dimension 22 " x 7 " x 1 ") is maneuverable with the vector thruster.
- The vehicle is quite robust and can be maneuvered under surface waves.

Acknowledgement.

The researchers (P. Ananthakrishnan and his graduate students who had worked on this project) wish to express their appreciation and gratitude to the Office of Naval Research for supporting the present work.

REFERENCES FOR SECTION 2.7

[1] J. V. Wehausen, *Ship Dynamics*, Lecture Notes, University of California at Berkeley, Fall 1972.

[2] J. N. Newman, *Marine Hydrodynamics*, The MIT Press, 1999.

[3] P. Ananthakrishnan and Sophie Decron, ``Dynamics of Small- and Mini-Autonomous Underwater Vehicles: Part I. Analysis and Simulation for Midwater Applications", *Technical Report*, 63p, Department of Ocean Engineering, Florida Atlantic University, July 2000.

2.8 Chemical Sensors
PI: Dr. Richard Granata

2.8.1 Summary

This section describes the formulation of a chemical method to detect underwater trace explosives, as well as the design and testing of a field-deployable device to implement the chemical method. The research goals are identified, the test materials, equipment and experiments are described and the results are discussed. The chemical compound, europium thenoyltrifluoroacetone, has been identified as an integral part of a viable underwater chemical detection method for underwater explosive traces. Included in this section is the final report on the capabilities of a chemical sensor UUV payload for detection of explosive materials for UUV applications.

2.8.2 Introduction

The ultimate purpose of this UUV component is to detect underwater explosives and provide a signal so that action can be taken. This process breaks down to three basic steps: (1) Obtain an underwater sample for testing, (2) Analyze the obtained sample and (3) Provide feedback of the results so that appropriate action can be taken.

Several methods exist to analyze a water sample for explosive traces [1], but practicality in UUV application dictates several limitations, such as size, cost, autonomy and processing speed. Consequently, these limitations in conjunction with the unique seawater environment eliminate most existing explosive detection methods. The research contained herein focuses on the formulation and testing of a detection method based on fluorescent marking and the development of a field-deployable device to detect waterborne explosive traces with this method. Attention has been given to UUV parameters such as size, cost, power consumption, autonomy, analysis speed and sensitivity.

The research goals have been identified as follows:

- Evaluate the feasibility of developing a photoluminescent method of detecting underwater explosive traces.
- Examine different fluorescent compounds, looking for optimal combinations of the chosen fluorescer (europium) and sensitizing ligands to achieve both fluorescent loss in water (quenching) and maintained fluorescence in response to explosive compounds. Different combinations, concentrations and mixing orders of the chemicals are evaluated. Other factors that influence the performance of the compounds are also evaluated, such as the amount of solvent required and/or used to deliver the chemicals into the seawater solution.

- Characterize the excitation and emission frequencies of the sensitized compounds.
- Evaluate the hypothesis that europium complexes will preferentially bond with explosive compounds over water molecules in an aqueous environment. This experiment to emphasize a seawater environment.
- Set up a working underwater explosive detection device and develop and execute a test plan.

2.8.3 Methods, Assumptions, and Procedures

2.8.3.1 Primary Test Materials

The primary test materials include the explosive sample and the chemicals involved. Medical nitroglycerin (NG) tablets are used for the explosive sample to accommodate safety issues of the university. The method should be extendable to a wide range of explosive compounds based upon nitro chemistries. Europium is used for the fluorescer and two compounds were evaluated as sensitizing ligands: Thenoyltrifluoroacetone (TTA), ($C_8H_5F_3O_2S$) and 1,10 Phenanthroline Monohydrate (OP), ($C_{12}H_8N_2 \bullet H_2O$). The TTA material was selected as the preferred ligand based on better fluorescence performance and environmental considerations.

2.8.3.2 Primary Test Equipment

For the first stage of laboratory testing, the primary test equipment consisted of a handheld UV light (approximately 370 nm) and a Perkin-Elmer LS50B luminescence spectrometer. The handheld UV light was used to execute preliminary evaluations of different chemical mixtures under different conditions. This provided a quick, efficient method of testing the design path, without performing tedious, exact experiments for all possibilities. The luminescence spectrometer was used to precisely evaluate certain mixtures for fluorescence and quenching. The luminescence spectrometer can either record the light output of a compound with a given excitation wavelength, or it can scan for the best excitation wavelength to produce the maximum intensity of a given emission wavelength.

For the second stage of testing, the focus is on a working field detector design. The core of the design is a compact, underwater fluorometer (Figure 2.8.1). The fluorometer used in this experiment is a WET Star model, made by Wet Labs, Inc., that has been specially modified to provide 370 nm excitation and record 613 nm emission. Due to the low velocity, laminar flow that is fed into the fluorometer, an in-line static mixer is also utilized to assure proper mixing of the seawater and reagent solutions.

Figure 2.8.1 – WET Star Fluorometer.

2.8.3.3 Experiments

The luminescence spectrometer was used to evaluate the appropriate excitation and emission wavelengths for the chosen fluorescent compounds. Special attention was given in determining the excitation wavelength so that it corresponded to a standard, commercially available, LED ultraviolet light source. This consideration was included so that an LED light source could be used in the field-deployable device. Background fluorescence analyses were conducted for several solutions to provide additional insight into the real fluorescence change between explosive-laden and explosive-absent solutions. Several fluorescent compounds were compared to determine the best choice of sensitizing ligands, concentrations and mixing orders. The effect of the solvent that was used to deliver the chemicals into the solution was evaluated for its effect on the explosive detecting ability. The detection limit of nitroglycerin in the luminescence spectrometer of the chosen compound was evaluated. The performance difference between seawater and fresh water was evaluated to determine if the additional constituents of seawater affect the detection method. The customized LED spectrometer was ordered, its performance evaluated in the specially fabricated vehicle mount and tested in the operating vehicle.

2.8.4 Results and Discussion

The following section is taken from the completed thesis conclusions section [2] which summarizes the finding of this study. The completed thesis [2] and open literature

manuscripts [2b,2c] are not publicly accessible pending outcome of intellectual property determinations.

It was determined that the use of a lanthanide element to fluorescently mark explosive traces is a viable underwater trace explosive detection method. While water quenches europium compound fluorescence, water-borne nitroglycerin is able to protect europium's fluorescent properties. This likely occurs because the explosive trace's negatively charged nitrate moiety is more strongly attracted to the positively charged lanthanide ion's free bonding site than are dipolar water molecules.

To capture the fluorescent properties of a lanthanide ion, radiation-absorbent ligands must be attached to absorb and transfer energy to it. The type of ligand is important, as well as mixing order if multiple ligands are used. It was found that the europium / thenoyltrifluoroacetone (Eu/TTA) complex produced significantly better results in underwater explosive detection than europium / thenoyltrifluoroacetone / 1,10-phenanthroline (Eu/TTA/OP) and europium / 1,10-phenanthroline / thenoyltrifluoroacetone (Eu/OP/TTA) complexes. Eu/TTA fluoresced strongly in the presence of NG, but almost completely lost fluorescence when NG was absent. On the other hand, Eu/OP/TTA and Eu/TTA/OP fluoresced strongly with and without water-borne NG. This suggests that the OP ligand creates a hydrophobic environment around the europium ion, even when NG is not present. The presence of the OP ligand also significantly reduced the solubility of the compound in methanol. Additionally, while Eu/TTA/OP and Eu/OP/TTA solutions contained the same ratios of components, they performed differently, indicating the importance of ligand mixing order.

It was found that the excitation wavelength required to create fluorescence of a lanthanide compound depended strictly on the excitation wavelengths of the attached ligands. When the TTA ligand was used, optimal excitation was found to be 382 nm and when the OP ligand was added, strong excitation also occurred around 310 nm. Excitation near the TTA requirement is easily accomplished via LED sources, whereas the deep ultraviolet wavelengths required by OP are not. Because of this and the better explosive-detection performance without OP, OP was omitted to provide an optimum compound for use. Since the thesis is ultimately aimed at a working design, practicality was factored in and excitation was chosen to be 370 nm for experimentation, versus the optimum wavelength of 382 nm. This choice was made because 370 nm is a standard wavelength available in LED's. To verify the correctness of this choice, testing was conducted on the Eu/TTA compound with both 370 nm and 382 nm excitation wavelengths for comparison, which indicated that very little performance is lost by this shift in excitation. Even less loss is expected in the field due to the fact that the 370 nm and 382 nm gap is closed somewhat due to the actual width of each one's excitation peak.

It is sometimes possible for the characteristic emission wavelength of an element to shift when it is combined with other components to form a compound. It was found that the characteristic europium emission wavelength of 613 nm persisted, regardless of the compound configuration. This wavelength did not change in the presence of OP, TTA, nitroglycerin or sodium, or in fresh water and seawater solutions.

Because europium fluorescence is quenched by water, it was necessary to combine the europium and sensitizing ligands in methanol before introduction into the seawater and water solutions. It was found that the methanol affects both the final solution clarity and fluorescence. Overall, the less methanol included, the better was the performance. For the tests conducted with Eu/TTA, fluorescence fell to negligible levels when the methanol level reached 35 percent of the total solution. Only Eu/TTA was tested for methanol effect because it was chosen as the more favorable compound in an earlier test. Rough solubility limits of the compounds were ascertained to provide some insight into the minimum amount of methanol required. OP had a negative effect on solubility. The maximum solubilities found for Eu/TTA, Eu/TTA/OP and Eu/OP/TTA were 1.02×10^{-2} M, were 4.57×10^{-3} M and were 4.53×10^{-3} M, respectively.

The europium detection method was found to perform considerably better in fresh water than in seawater. A specified amount of nitroglycerin could be detected in fresh water with less than 1/12 the amount of reagent required to detect the same amount of nitroglycerin in seawater. Based on references [3-6], it is believed that this is due to metal-exchange reactions with calcium and magnesium in the seawater. References [3-6] also note that acidic conditions negatively affect europium compound fluorescence. The impact of metal-exchange reactions and low pH were not quantified because the calcium and magnesium content of seawater is not expected to vary significantly from the seawater samples used for experimentation and the range of seawater pH is much higher than the problem ranges reported in references [3-6]. However, tests were conducted to prove that this explosive detection method is susceptible to these conditions and help explain the difference in performance between seawater and freshwater. These tests confirmed that this detection method is compromised by large amounts of calcium ions and low pH.

With Eu/TTA at $1x10^{-4}$ M concentration (total solution), nitroglycerin could be detected in the laboratory luminescence spectrometer down to concentrations as dilute as approximately $1x10^{-6}$ M.

After characterizing the chemical detection method in the laboratory with a luminescence spectrometer, tests were performed with a modified commercial fluorometer to move towards a field-deployable design. Static (non-flowing) tests indicated that, with this chemical detection method, a deployable fluorometer is sensitive to nitroglycerin dissolved in seawater. The sensitivity depends on the amount of the europium complex used, with more Eu/TTA translating to better sensitivity. In the WET Star characterization tests, sensitivity was found to be 2.44×10^{-7} M nitroglycerin with the equipment used, a Eu/TTA concentration in methanol of 4×10^{-4} M, and a mixing ratio of 8 percent. These concentrations convert to 28 ppb. However, there is a limit to which the Eu/TTA concentration can be increased before problems are encountered with the particular fluorometer used in this experiment (WET Star). If the Eu/TTA concentration is high enough that the upper output voltage limit (5 V) of the WET Star was surpassed, the WET Star output information was inconsistent with visual observation and luminescence spectrometer readings. At these high Eu/TTA concentrations, the WET

Star indicated that there was less intense fluorescence with nitroglycerin than without, even though it was visually obvious that the opposite was true. Based on these comparisons, it was concluded that the WET Star output was erroneous when the Eu/TTA concentration was too high. Therefore, the best performance with this method and equipment is attained when the Eu/TTA complex is as high as possible, without reaching the point where the fluorometer outputs false results (possible off-scale digital-analog conversion). While higher europium complex concentrations bring better sensitivity (before saturation), they also require more reaction time. Until the reaction is completed, the fluorescence output oscillates erratically and produces little usable information. All of the concentrations studied needed less than five minutes to stabilize. Reaction time must also be considered in system design.

The impact of sample filtration was also addressed, and it was found that filtration slightly increases the fluorescence intensity reading from the fluorometer. This slight increase was noted in both the nitroglycerin-laden and nitroglycerin-absent solutions, with very little change in their relative readings. With minimal change in fluorometer output and no noticeable change in relative readings, filtration adds little value to the design. However, if a pump is used to pass the sample through the fluorometer, a minimum amount of filtration will be required to assure pump operation and endurance.

The flow-through trace-explosive detector design was validated with a laboratory hydraulic system. This system combined the seawater/nitroglycerin solution with the europium complex solution in an appropriate ratio and then mixed them, before the final solution was passed through the modified WET Star fluorometer. Using this system, the fluorometer was able to discriminate between plain seawater and seawater that contained traces of nitroglycerin, and the design concept was proven.

It is believed that the negatively charged nitrite moiety of the nitroglycerin compound is what makes it detectable with the chemical method presented herein. Because this characteristic is common to many explosive types, it is believed to be highly likely that this method can be extended to detect many explosive types, in addition to nitroglycerin.

Based on this research, two proposed design options are shown below in Figures 2.8.2 and 2.8.3. The first design utilizes two small pumps, while the second makes use of one pump and a restrictor combination to control the seawater / reagent ratio. The UUV speed cannot be assumed to be constant, and because the mixing ratio of the seawater and reagent must be controlled, at least one pump is necessary. The two-pump design would be easier to setup, while some tuning would be required to achieve the proper mixing ratio with the restrictor setup. The restrictor setup would be less expensive and likely require less maintenance. Cursory research indicates that pumps and restrictors are available that meet the requirements of this design. For example, Micropump, Inc., can provide suitable pumps, and The Lee Company produces a range of hydraulic restrictor sizes that will fit this application. Many companies make small pumps, but this application is quite demanding for miniature pumps. The pumps must be accurate in their flow rates and more importantly; they must be able to withstand the internal case pressure that results from water depths that the CCST UUV must be designed to. Static

mixers are available from a variety of companies. TAH Industries provided the static mixer used in the proof of design test of this thesis. Work proceeded with the two pump design (Figure 2.8.2).

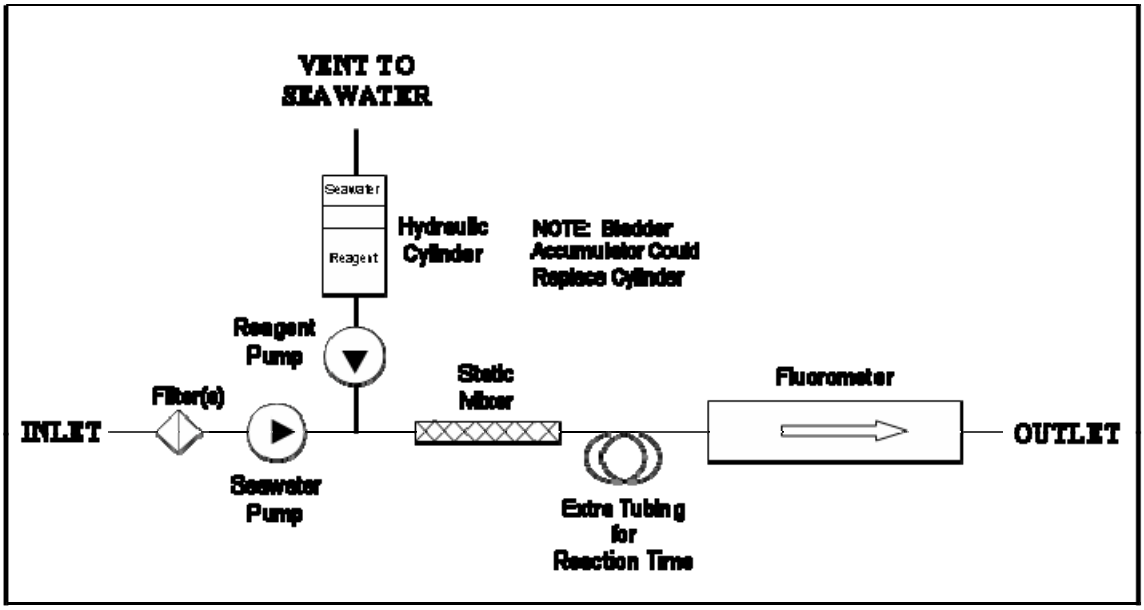


Figure 2.8.2 – Proposed design schematic no. 1

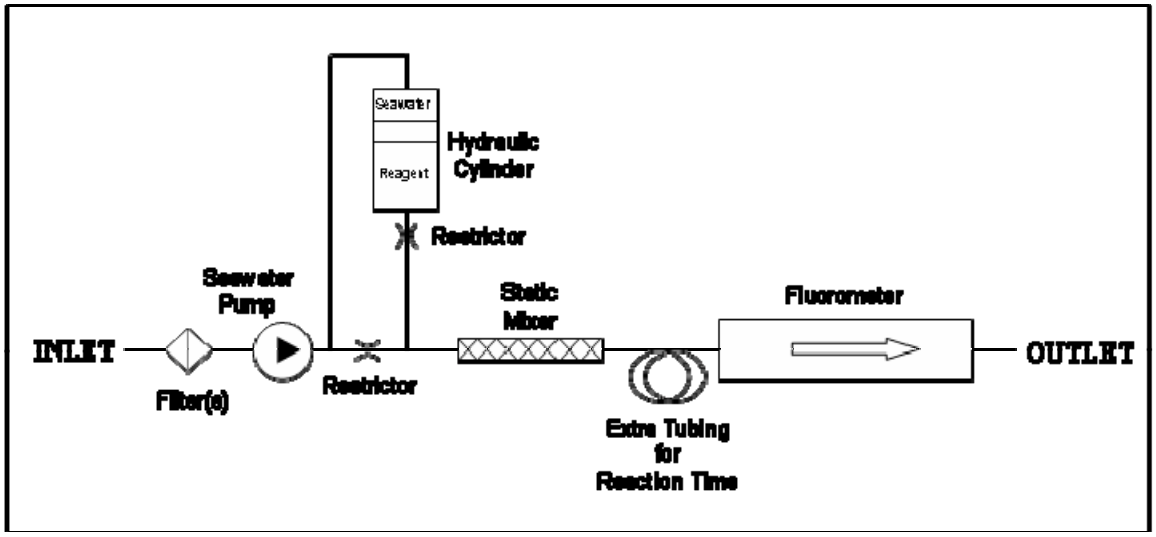


Figure 2.8.3 – Proposed design schematic no. 2

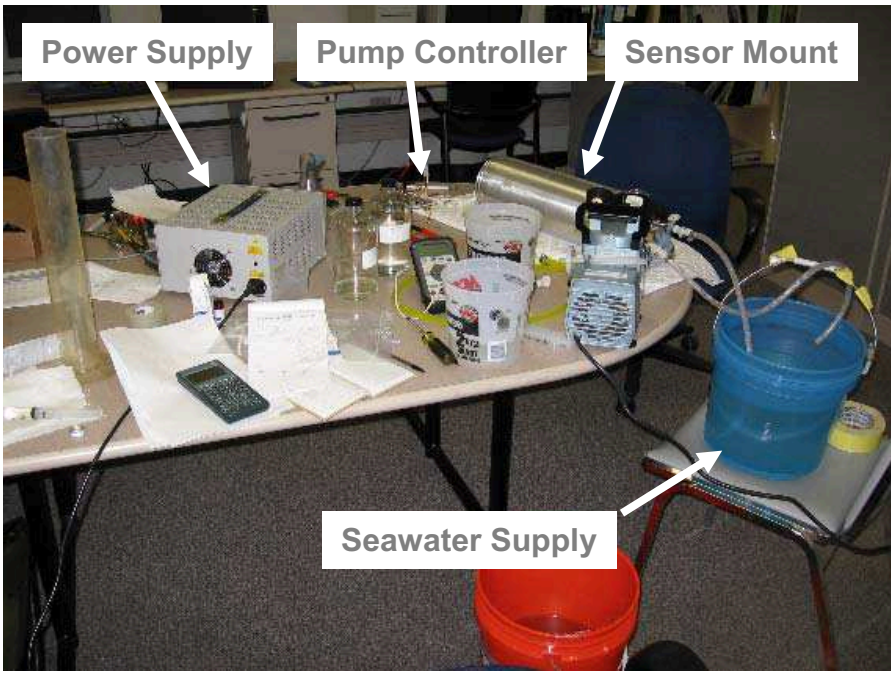


Figure 2.8.4 – Laboratory test setup of chemical sensor installed in the vehicle mount.

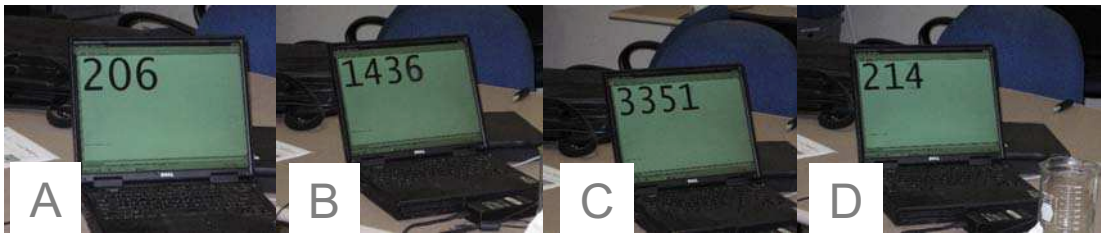


Figure 2.8.5 – Laboratory test setup of chemical sensor installed in the vehicle mount. Readings obtained were: A) seawater plus methanol (baseline); B) seawater plus reagent in methanol; C) seawater with nitroglycerin plus reagent in methanol; D) seawater plus methanol (baseline). The concentrations were: a) nitroglycerin in seawater ($5x10^{-4}$ M) and, b) reagent in methanol ($4x10^{-4}$ M). The mixing ratio was 9:1 (a:b) in image C.

2.8.5 Conclusions and Recommendations

From this work, it was determined that the use of a lanthanide element to fluorescently mark explosive traces is a viable underwater trace explosive detection method. Europium was used as the lanthanide element. While water quenches (shortens) the europium compound's fluorescence, water-borne nitroglycerin enhances (prolongs) its fluorescence.

To capture the fluorescent properties of a lanthanide ion, radiation absorbent ligands must be attached to absorb and transfer energy to it [7]. The type of ligand is important, as well as mixing order if multiple ligands are used. Thenoyltrifluoroacetone (TTA) is a good ligand to use with europium for this purpose. Ortho-phenanthroline (OP) is not recommended because its absorption range does not coincide with that attainable with LED light sources and it appears to prevent fluorescence quenching when explosive traces are not present.

The combination of europium and TTA is recommended as a compound to detect underwater explosive traces. It is also recommended to limit the amount of chemical solvent (methanol) to as low a percentage as practicable, definitely not to exceed 20%.

Experiments have been completed to allow selection of final chemical detection method. Also, a sensor module was identified, modifications specified, the UUV operable hardware ordered and received. Design, fabrication, assembly and testing of the sensor module vehicle mount has been completed. The subsequent phase of the work is also complete which is laboratory and field testing of the chemical sensor module installed within the UUV. Figure 2.8.4 is an image of the laboratory test setup.

Using the laboratory test setup, detection of nitroglycerin was verified when bypassing the reagent storage bladders. This was replicated in three other tests. However, when the bladders were filled and used as the reagent source (same solutions) detection could not be verified. High fluorescence readings were obtained with and without nitroglycerin. There appears to be an issue with the bladders. There are several possibilities:

1) The bladders may affect the flow rate of the reagent pump. *Small possibility.*
2) The leaching that we observed from the latex bladders may affect the reagent solution. *Most likely possibility.* According to chemical and material compatibility guides, latex and methanol should be fully compatible. Also, the manufacturer of the bladders has indicated that methanol should be compatible. Despite these assurances, the latex material has been observed to discolor the reagent solution and allow it to pass through the bladder walls. It could be that this latex material is not fully compatible with the methanol, or the other chemicals within the reagent solution (Eu and TTA) have an adverse affect on the bladder material.
3) There may be pockets within the bladders may not fully evacuate, resulting in concentrations that are different than what is expected. *Slight possibility.*
4) The chemicals may be building up in the bladder system by adsorption or entrapment in geometric irregularities. *Slight possibility.*

The bladder manufacturer has refused to disclose additives or curing agents that may affect our results, but has disclosed that calcium nitrate was used as a coagulant in the process. Calcium nitrate would not account for the discoloration of the reagent solution, but would have strongly affected the fluorescence response being tested.

Figure 2.8.5 shows images of the sensor output in the laboratory tests with the latex reagent reservoirs (bladders) bypassed. The latex bladders were bypassed because they were the likely source of contaminants that caused sensor output problems. The serial

output readings from the fluorometer shown in Figure 2.8.5 were typical of those expected for the chemical sensor operated under the conditions specified (no reagent, with reagent, with reagent and nitroglycerin, and no reagent) demonstrating a complete cycle of readings from baseline through detection and back to baseline. The chemical sensor in its vehicle mount was tested in the UUV, which showed responses related to baseline and contamination from the latex bladder. The bladder material requires improved resistance to methanol reagent.

Despite the issue with the bladders, the chemical sensor have been a success overall. The reagent reservoir is an issue that could be corrected, and it does not change the fundamentals of the detection method and device. The two main accomplishments have been:

1) A chemical method has been identified that has been proven to repeatedly identify nitroglycerin. This identification can also be made to obvious to the naked eye. The fundamental idea is that the chemical method works.
2) A device to implement the chemical method has been shown to work (aside from the problems with the reagent reservoir). When using a cup as a reagent reservoir, instead of the bladders, this device has been used to detect nitroglycerin four separate times between November and April, with similar results each time. The fundamental idea is that the chemical method can be incorporated into a detection device and operated within a UUV.

2.8.6 References for Section 2.8 - Chemical Detector

1) J. Yinon, S. Zitrin, Modern Methods and Applications in Analysis of Explosives, John Wiley & Sons Ltd., 1993.
2) T.A. Langston, "Chemical Method and Device to Detect Underwater Trace Explosives via Photo-Luminescence," M.S. Thesis, Florida Atlantic University, December, 2006.
3) C. N. Shtykov, T. D. Smirnova, Y. V. Molchanova, Synergistic Effects in the Europium(III)-thenoyltrifluoroacetone-1,10-Phenanthroline System in Micelles of Block Copolymers of Nonionic Surfactants and Their Analytical Applications, Chernyshevsky State University, Saratov, Russia, January 2001.
4) A. Adeyiga, P. Harlow, L. Vallarino, and R. Leif, Advances in the development of lanthanide macrocyclic complexes as luminescent bio-markers, Department of Chemistry, Virginia Commonwealth University, 2006.
5) Perkin Elmer Life Sciences, Stability of the Wallac LANCETM Eu-chelates, LANCETM Time-Resolved Fluorescence Detection Application Note.
6) Cisbio International, New Europium Cryptates to Probe Molecular Interactions Using HTRF.
7) E. Menzel, K. Bouldin, R. Murdock, Trace Explosives Detection by Photoluminescence, TheScientificWorld JOURNAL (2004) 4, 55–66 ISSN 1537-744X; DOI 10.1100/tsw.2004.7.

3.0 HIGH DEFINITION VIDEO SYSTEMS

PI: Dr. William Glenn

Tasks 3.24-3.27

3.1. Summary

During year 3 the Florida Atlantic University Imaging Technology Center achieved all of its objectives under this program with the completion of the video compression system and solid-state recorder and integration of the final system. The overall goal was the successful demonstration of a 2160-line, progressive-scan, ultra-high-definition 3D imaging system that combines a pair of FAU's HD-MAX video cameras with a pair of Sony SRX-R105 digital cinema projectors for stereo imaging and projection. Included in the system were a solid-state recorder and a video compression/decompression system design. All items under design and development were completed and made ready for delivery or demonstration, as required by the sponsor. In addition, special reports were prepared detailing different aspects of the system, including polarization optics, camera and projector setup, and JPEG-2000-based video compression processor design.

3.2. Hardware Design, Fabrication, and Testing

The following tasks were successfully performed.

Cameras

- Modification of the cameras to allow synchronization of a camera pair
- Design and of an adjustable mount for a stereo camera pair

Video Compression/Decompression System

- Completion of the video compression/decompression circuitry design
- Completion of tests of the Analog Devices ADV212 JPEG 2000 Video Codec compressor chip, a replacement chip from Analog Devices that corrects defects of the ADV202 originally selected for the design.
- Final assembly and testing of the video compression/decompression system.

Solid State Recorder

- Successful conversion of the record portion of the solid-state-recorder (SSR) from interlaced to progressive scan at 30 frames per second
- Testing and debugging of the SSR in synchronized-pair mode. The SSR operates flawlessly now in record mode with the 4GB flash memory cards installed, but minor timing problems persist on playback that introduce minor flicker in the projected images.
- Both pair's of SSR were successfully tested for 8 minutes of record and playback.

Support Instrumentation

- Design and assembly of the dual power supply

3-D Imaging

- Polarization control experiments were conducted to determine whether the right- and left-view images from the two Sony SRX projectors could be coded via circular polarization. It was concluded that the internal design of the Sony projectors prevented this method from being implemented without introducing significant bleed-through between right and left images, and linear polarization was selected instead, at the cost of a reduction of luminance levels at the projection screen.
- The polarization-preserving projection screen was received and installed.
- Two Quad HD cameras equipped with 50mm lenses were mounted side by side and polarization-coded 3-D imagery was projected onto the screen with good results.
- Thickness variations in the linear polarizers were found to introduce aberrations in the imaging system with a subsequent reduction in image sharpness. Thinner, optically flatter polarizing film was ordered and tested with positive results.
- Expected differences in the psychological impact of the 3D imagery were observed when camera separation and angular orientation were changed.
- The 3-D imaging system, combining two cameras with the two projectors, was tested on multiple occasions with different groups of people as observers.

Instructive Comments on Ultra-Broadband Video Imaging & Image Processing System Design Made by W. E. Glenn in Discussion with Navy Lab Personnel

The most important information in surveillance is in the images of fast moving objects. The faster they go the greater the threat and the more necessary it is to have real time computer processing. Slow moving objects can be analyzed by eye and there is lots of time to respond. For fast moving objects (missals, airplanes, fast moving trucks) you don't have time to analyze your entire field of view before it is too late. Real time processing at high resolution and at least 30 frames per second will be essential for response to threats.

The camera output is SMPTE 292 standard and the display adapter output is a DVI standard. They are both at 3 gigabits per second.

A digital camera with over 10 million pixels bought at a camera store would cost a lot less than the HDMAX camera. It would have the same resolution. It would not be at 30 frames per second. The commercial camera can put out still images that can easily be transmitted and they interface easily with a computer for non-real time processing.

The real advantage of the HDMAX camera is the 30 frame per second frame rate. This is very important for fast moving objects. However, the computing and transmission problem would have to be solved before a test can be performed on fast moving objects. Real time viewing with transmission over a fiber link is the best you can do until those problems are solved.

The sensor is 3840X2160 pixels progressively scanned at 30 frames per second. It has an electronic shutter.

The camera output is in two coax cables at 1.5 gigabits per second (SMPTE 292)each.

The camera output has two feeds to a display adapter processor that reformats the signals into 8 vertical stripes (2160X480). This is fed to the display on DVI cables. The

frame rate is still 30 frames per second. The drive for each sector is 375 megabits per second.

The camera uses standard 35mm camera lenses which can be either fixed focus or zoom. We are using Canon and Panavision lenses.

Our camera is unique in that it can provide 4K resolution at 30 frames per second. Lower resolution or lower frame rates are available in inexpensive commercial cameras. I gather you would like to have the capability of transmitting the output of the camera over a wireless connection and process the information in real time at its destination. There are two bottlenecks to doing this at this time. The wireless link cannot handle the bit rate. We can build a compression encoder and a decoder that can reduce the transmitted bit rate to about 100 megabits per second. If your wireless link can handle that then that would solve that problem. The other problem is that the computers that are used to process the information are too slow and cannot process in real time regardless of the reformatting. There are at least two commercial processors that can process at those bit rates. They are made by AMBRIC and by SRC Computers. It would be a good idea to check with the company doing your processing to see if they can program one of those computers to process the information in real time.

A standard computer cannot accept and process information at this bit rate in real time regardless of how it is formatted. The suggested technique simply frame grabs a sequence and processes off line in non-real time. It takes much faster processing hardware to process this in real time. This can be done by making your own computer with FPGAs. A lot of companies do this (Mostly in Japan or Holland but a few in the US).

The military has funded IR sensors for decades but has not funded high speed, real time processing hardware. For surveillance this is very important. You would like to have an image of everything in your field of view with high resolution. The important objects are those that are moving. (If it doesn't move it is not a threat). The faster it moves the more serious the threat. A missile is worse than a low flying airplane, is worse than a speed boat or truck, is worse than a man running, is worse than a man walking etc. The faster the motion the higher frame rate is required to do the processing. A man cannot process the information at that resolution and at that speed. It must be done with high resolution, high frame rate imagers and high speed processors. The Japanese government laboratory, NHK, is processing 33 megapixel images at 60 frames per second. The processor accepts 24 Gigabits per second bit rate.

There are laboratories in the US that are capable of building high speed digital processors. Our laboratory can and so can Imperx and Avid. It may be possible to build a hybrid system to do the job. Our display adapter processor can process with an input up to 4.5 gigabits per second. It could be programmed to do moving target indication of moving parts of an image. It could tell you the speed and direction of motion of an object. It could do an electronic zoom on the object and send a low resolution image of the moving object to a computer at 30 frames per second. This would be a low enough bit rate for further analysis by the computer. If two cameras were used to produce a stereoscopic pair the distance of the object could also be determined. A missile at the horizon looks about the same as a pidgin up close. You don't want to waste Patriot missiles shooting at pidgins.

Processing the compressed signal would be a real bear programming wise. I have been assuming that we would decompress back to video at the receiver. This is now 3Gbits/second. The programs you have developed on your computer can be implemented in an FPGA at our bit rates. That means building a processor and programming it. The program in the FPGA can be changed if you need to. You just have to have one big enough for the program. You might also need some external DRAM memory. Our display adapter has two 4 million gate FPGAs and some DRAM running at these rates. Since I don't know what your computer processing does, I have no idea whether it could do your process if it was reprogrammed.

Demonstrating the camera with a fiber or coax link to our 4K LCD display can be done at any time. Compressing the signal and interfacing with your computer system will take a lot longer (six months or so). We have a compression board completed but not programmed yet. For you to use the system for transmission we will need to duplicate the board so that the received signal can be decoded back to two 292 signals. From there we can display it on our 4K LCD display. Just to give you an idea of the transmission bit rate after compression:

At 300Mbits/second the reconstructed image will look just like the original

At 150Mbits/second there will be slight image artifacts.

The compression system is programmable to give different compression ratios. You would not want images compressed below 75Mbits/second.

The decompressed signal will be back at 3Gbits/second. We are trying to find out what bit rate CAT5 can handle. Also what bit rate can the computer handle? We have been doing all of our processing by programming FPGAs. They can process at this bit rate. They are available with a very large number of gates. We are using two 4 million gate FPGAs on a 22 layer PC board. Chips of this type have been made experimentally up to two billion gates per chip. For high speed computing in real time this is the way to go. NHK in Japan is now processing 8K video at 60FPS. This is at 24Gbits/second.

Specifications for the Single Monochrome Camera:

The items with * will be needed to display and record the full resolution output

1. 3840×2160 CMOS sensor
2. Single sensor Bayer pattern color CMOS
3. Continuously variable frame rate up to 30 FPS
4. Electronic shutter from 14 microseconds to 1 second
5. Sony 4K theatre projector*
6. Projector displays in interlaced format. Optional LCD display displays in progressive format
7. 12 bit encoding before gamma- 10 bit encoding after gamma.
8. 33MM diagonal active frame (Uses standard 35MM cinema camera lens)
9. Programmable gamma

10. Auto black level
11. Auto white balance
12. Bad pixel correction
13. Shading correction
14. FPN correction
15. Programmable color balance
16. 1080 line output progressive or interlaced
17. 1080 line frame grab output for computer download
18. 6" LCD color viewfinder.
19. Programable color matrix on 1080 line signal
20. Menu on viewfinder for camera controls
21. Remote digital camera control.
22. Two SMPTE 292 serial digital outputs. One output in SMPTE 274M 60 I format. The sum of the two (SMPTE 372M) produces 274M 30 P format. Also 3840X2160 Beyer pattern signal.

In the display module: *
1. 3840X2160 color output to interface with Sony 4K theatre projector
2. Conversion from Beyer pattern to 3840X2160
3. Programmable color matrix
4. Frame grabber to download 3840X2160 image to computer.

Display: *

Sony 3840X2400 color LCOS theatre projector

Camera head power consumption: about 30 watts at 24 volts.
Viewfinder power consumption: 18 watts at 24 volts.
Camera size 2.5"X4.5"X9"
Weight without lens 3.75 pounds.

Several measurements have been made on the camera. The sensor was exposed to high energy protons with an exposure equivalent to one year's exposure in the space station. Pixel defects after exposure less than one per million pixels. The bad pixel correction should handle a life of at least 10 years at high altitude.

By removing the IR filter the camera becomes sensitive to the near infra red since the color filters are transparent to infra red. In this condition it has the full monochrome 3840X2160 resolution. The sensitivity is much higher than to visible light.

The S/N ratio without gamma and after FPN correction is approximately 64db.

Camera and Projector Figures

The HDMAX camera pair is shown in Fig. 1, the Sony SRX projector in Fig. 2. All systems are now ready for delivery.

Figure. 3.1. The pair of color HDMAX cameras used for the 3D video imaging system.

Figure. 3.2. The Sony SRX-R105 digital cinema projector, one of two used for the 3D video display system.

3.3. Polarization Control for 3-D Imaging with the Sony SRX-R105 Digital Cinema Projectors

Separation of right- and left-eye images is generally achieved in 3-D movie theaters by means of light polarization. The Sony SRX-R105 digital cinema projectors used by Florida Atlantic University for display of 3-D images produced by a pair of HDMAX ultra-high-definition video cameras produce polarized outputs, but with the green component polarization orthogonal to that of the red and blue components. Furthermore, the absence of appropriate color trimming filters in the projectors introduces some crosstalk between the color components. For these reasons we chose to encode the two images by means of conventional linear polarization screens positioned in the paths of the two output beams from the projectors.

Although there are alternatives, 3-D movie theaters usually rely on polarization to code the right- and left-view images projected on the theater screen. The local Muvico theater in Boca Raton, Florida, for example, rapidly alternates the right- and left-view images for 3-D movies, coding one with right-circularly polarized light and the other with left-circularly polarized light. Viewing glasses worn by the patrons select the correct polarization for the correct eye. Also commonly used are horizontal and vertical and +45° and -45° linear polarization. Circular polarization has the minor advantage that a viewer can tilt his or her head with no change in the relative luminance of the right- and left-view images received at the eye. With linear polarization, a side-to-side tip of the head introduces changes in image luminance and some left-right cross-talk, i.e., the right eye sees some of the left-view image and vice versa.

The output of the Sony SRX-R105 digital cinema projector is inherently circularly polarized, and we thought initially that we would exploit this characteristic and code left- and right-view images using circular polarization. Unfortunately, whereas the red and blue components of the output of the projectors are right-circularly polarized, the green components are left-circularly polarized. This difference arises because beam combiners in the projector for the red and blue components introduce one fewer reflection than do the beam combiners for the green image component, a reflection reversing the sense of circular polarization. Because of this complication, along with other aspects of the SRX-R105 system, we ultimately chose linear polarization for coding the two images.

Were all of the light output of a given projector circularly polarized with the same sense—i.e., RC or LC—right and left image coding would, at least in principle, be quite simple: Light that is RC polarized can be changed to LC-polarized light by passing it though a half-wave plate, a thin plate of birefringent material made in such a way that light of one linear polarization is delayed by 180° (corresponding to a retardation of an odd multiple of one-half wavelengths) relative to light of the other linear polarization. Thus, the light output from one projector could be RC and that of the other could be converted by the half-wave plate to LC. This operation would not be perfect, however, because a birefringent wave plate that acts like a half-wave plate at one wavelength does not necessarily behave like a half-wave plate at other wavelengths. Indeed, even the best half-wave plate will convert RC to LC only at a single wavelength: at other wavelengths the conversion is from RC to some more general form of elliptical polarization. For wave plates operating in the first order, where the retardation is exactly □/2 and not, e.g., 3□/2 or 5□/2, this wavelength dependence may be completely negligible for light with a

spectral bandwidth of 100 nm or less but not necessarily for white light. A wave plate operating in a high order, e.g., 15□/2, could behave like a half-wave plate at 500 nm but like a quarter wave plate at 300 nm and 700 nm, converting the light at those wavelengths from RC to linear polarization.

Were it possible to access the optical components inside the SRX-R105, good results could probably still be achieved. The scheme to be employed in that case would be to insert a wave plate in the green-light channel that was a half-wave plate at the mid-band wavelength for that channel and exhibited negligible dispersion in the long- and short-wavelength tails of the green-light spectral distribution. The green light would thus be converted from LC to RC, and all three spectral components of the projected image would have the same polarization. In the second projector, the opposite objective could be achieved by placing half-wave plates in the red- and blue-image paths, converting the polarization of those image components from RC to LC. Because the red-, green-, and blue-component spectral bandwidths are relatively small, such a scheme should work well.

If access to the inside of the projectors is precluded—in our case by warranty restrictions—a scheme developed by ColorLink can be employed, though with a more complicated system of polarization-control components and with somewhat poorer results. In this case, a wave plate intercepting the output of a projector is selected to behave—approximately, for that is all that can bee achieved—like a half-wave plate for the mid-band green light and like a full-wave plate (i.e., it has no effect) for the red and blue component light. This conversion is effected for both projectors. The output of one of the projectors is then passed through an additional wave plate that behaves like a half-wave plate over the entire visible spectrum. The result is one projector with, nominally, all RC-polarized light and the other projector with, again nominally, all LC-polarized light. Papers describing this scheme are to be found at ColorLink's web site: http://www.colorlink.com. If improvements described in the papers are also incorporated, this scheme works reasonably well, with relatively little insertion loss. We chose not to implement it because of uncertainties regarding the quality of the wave plates we could obtain—the goodness of anti-reflection coatings, the amount of cross-talk introduced as a consequence of the wavelength-dependency of the wave plate operation, etc.—and because of other aspects of the Sony projector design that reduced the degree of polarization of the three color components..

At one stage we considered another, conceptually simpler scheme for encoding the right- and left-eye images by RC and LC polarized light. The idea was to exchange the green-component images electrically, in effect by crossing wires, and to place a broadband half-wave plate in the output beam of one of the two projectors. This approach would, in principle at least, achieve the desired goal: all three components—red, green, and blue—of one image would be encoded by RC polarization and all three components of the other image by LC polarization. For this scheme to work correctly, it is essential that the images projected by the two projectors be in perfect registration; otherwise, a white object at a particular depth would be white in the middle, magenta at one end, and green at the opposite end, a clearly objectionable condition. We ultimately abandoned the scheme, not because we could not achieve the desirable level of image registration but because of other problems inherent in the polarization control of the SRX-R105. For

reasons we associate with deficiencies in the design of the SRX-R105, some magenta light reaches the green spatial light modulator and, similarly, some green light reaches the red and blue spatial light modulators. The effect is a reduction in the saturation of the colors that is noticeable in normal use of the projectors and the introduction of noticeable crosstalk between the left- and right-eye images if the green signal-swapping scheme is used.

After investigating all possible solutions to the polarization coding problem, we ultimately decided on the simple expedient of placing linear polarizers—one with transmission axis at +45° to the vertical and the other at -45°—in front of the two projectors. Polarized glasses designed to select out these two components were then purchased for viewing the 3-D video images. With this scheme employed, somewhat more than half of the output power of the projectors is lost through absorption in the linear polarizers. Nevertheless, even with the comparatively low-power SRX-R105 (the SRX-R110, rated at 10,000 lumens, produces twice the output of the SRX-R105), we found the resulting images to be of satisfactory brightness when projected on the large polarization-preserving screen some 40 feet distant. When the video image is of particularly high contrast, there is some "bleed through" of the right image into the left and vice versa. In general, however, this defect is not objectionable, and it would likely be observed with any of the alternative polarization-based schemes, too.

Our difficulties with polarization control suggest that Sony could do a better job in the design of their digital cinema projectors should they ever re-address that issue. In particular, they might look at the use of wave plates internal to the system that assure (a) that all three color components have the same circular polarization sense, RC or LC, and (b) that the polarization sense can be specified as an option in the purchase of a particular projector.

3.4. HDMAX Camera and Sony SRX-R105 Projector Configuration for 3D viewing: Typical Setup

In this report we provide a description of the HDMAX camera and Sony SRX-R105 projector configuration for 3D imaging as it is typically set up at in the Imaging Technology Center at Florida Atlantic University. This configuration gives pleasing results, although as discussed in a subsequent report, it does not provide full realism in the display because of discrepancies between the apparent distance to an object as determined by the angular subtense of the object and the apparent distance to the object as determined by the angular convergence of the observer's two eyes. In simple terms, it distorts depth somewhat. Basic configuration information is as follows.

Camera Positioning

The two HDMAX cameras, equipped with their standard 50-mm focal length Canon camera lenses, are positioned side-by-side and separated by approximately 4 inches. At this separation, the cameras will produce 3D imagery that exaggerates somewhat the depth of the scene as viewed on the projection screen. The cameras should both be level and at the same elevation. They are then aligned such that the optical axes of the cameras converge approximately 12 feet in front of the camera pair.

Any relative rotation of the cameras about their optical axes should be eliminated as much as possible: when one camera is level, the other should be too. Correcting misalignment of this kind can be most easily accomplished by first aligning the projectors using the built-in test patterns, as discussed below, and then projecting stereo-pair images with the two cameras capturing a scene that contains strong horizontal features. If the cameras are not both leveled to the same axis, the two images of a horizontal feature, viewed without the polarizing glasses, will appear tilted one with respect to the other. If one camera should be slightly rotated about its optical axis relative to the other camera, a swirl-like pattern may be observed on the projection screen if there is a richly-textured structure near the common plane of convergence such as grass. The basic idea is conveyed by Fig. 3, which shows a more-or-less random dot pattern overlayed on itself with perfect registration in (a) and with a slight rotation in (b). A relative rotation of the cameras about their optical axes can be compensated to a degree by proper adjustment of the screws that attach the cameras to the mounting plate by which the camera pair is attached to a tripod. Perfect adjustment is not necessary.

Figure. 3.3. Effect of camera rotation on projected overlay image: (a) with no relative rotation, (b) with several degrees of relative rotation.

The two Sony projectors are positioned one above the other, zoomed, focused, and adjusted such that the two test patterns produced by the built-in test-pattern generator are registered as well as possible. Vertical positioning of the images is accomplished by means of a remote-control-actuated servo mechanism that moves the digital light valve assembly up and down relative to the projection lens. (Details concerning the remote control for the SRX-R105 and other projector setup information are to be found in the Sony publication "SR Projector," Sony document 3-872-931-13 (1). Horizontal position of the images is accomplished by moving the projectors physically side to side and by angling them properly toward the projection screen. Relative tilt of one image with respect to the other can be corrected by means of the adjustable feet supporting the cameras. It is not critical that the overlaid images be in perfect registration for comfortable viewing of the 3D imagery—no super-human effort is required—but neither should they be significantly out of alignment. Vertical registration is more important than horizontal registration, since horizontal misregistration results simply in a change in apparent distance to a scene element, whereas vertical misregistration may preclude viewing of the scene at all: Note that for the highest possible definition, the optical axes of the projectors should be perpendicular to the projection screen (in most cases this means that the projector axes should be horizontal). The projector lenses are of

sufficiently high quality that it is possible for all individual image pixels to be in focus on a flat screen at a distance of 40 feet.

Some people have been confused on seeing our projector setup, expecting the projectors to be positioned side by side as are our eyes. In concept it does not matter whether the two projectors are mounted one above the other or side-by-side. What is important is that the two projected images be in registration. The vertical positioning servo mechanism works in such a way as to prevent keystoning of the images if the projectors are one above the other. Side-by-side positioning of the projectors makes it impossible to compensate for keystoning in the side-to-side direction (see Fig. 4), and some reduction in quality of the 3D display will result.

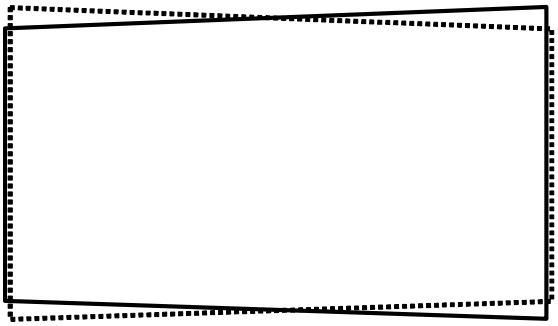


Figure 3.4. If the projectors are positioned side-by-side, the keystoning of one projected image (solid-line figure) will be opposite that of the other projected image (dashed-line figure), preventing good registration of the two images.

The polarization-preserving projection screen is 7'6" high by 16'4" wide. Best results are obtained if the projected image is zoomed to fill the screen in the vertical direction. The full image, having an aspect ratio of 9:16, will then be 7'6" high by 13'3" wide. Compared to other Sony digital cinema projectors, the SRX-R105 has relatively low output power—5000 lumens, compared to 10,000 and 20,000 lumens produced by the SRX-R110 and SRX-R120—but because of the gain (high reflectivity in the back-scatter direction) of the projection screen, the images are still satisfyingly bright for persons sitting in the usual viewing area.

For optimum viewing, observers should sit at a distance of about 12 feet from the screen. Under these conditions, the angle subtended by the picture will be as large as possible without individual pixels in the image being visible. Recall that it is the extremely wide angular field of view, coupled with the extremely high pixel count (resolution), that gives the 3D HDMAX imagery its great impact.

Since the distance from the observer to the projection screen is the same as the distance from the camera pair to the point of convergence of the camera axes, an object 12 feet in front of the camera pair will, according to parallax information received at the eyes, appear to the observer to be at that distance. Furthermore, because of the focal length of the camera lenses and the 40-foot projection distance, an object at that distance will appear in the image to have the correct size, i.e., a six-foot-tall person will appear to stand about six feet tall.

As noted above, the system configuration just described tends to enhance the perception of depth in the screen: front-to-back distances feel greater than they are in the actual world. For complete realism, the 3D theater depiction of a given object should display that object with a size and apparent distance—as determined by parallax—that are fully compatible. Consider two 6-foot-tall persons, one observed at a distance of 10 feet, the other at a distance of 30 feet. In the former case, the angular subtense of the subject from the perspective of the observer is $\tan^{-1}(6/10) = 0.540$ rad, and the convergence angle is $\tan^{-1}(2.5/120) = 0.021$ rad, assuming the eyes of the observer are separated by 2.5". For the case of the 6-foot-tall subject at 30 feet, the angels are, respectively, $\tan^{-1}(6/30) = 0.197$ rad and $\tan^{-1}(2.5/360) = 0.007$ rad. The display should duplicate these conditions.

Achieving such realism in the display is possible, but it requires more careful adjustment of the cameras and projectors than is suggested by the above steps. The principals underlying the achievement of such realism and means for their practical implementation are the subject of a separate technical report.

3.5. JPEG 2000 Compression Processor for Ultra-High Definition Recorder

The Imaging Technology Center at Florida Atlantic University has previously developed and demonstrated a solid-state digital video recorder (SSR) capable of recording and playing back ultra-high-definition video (UHD), i.e., 3840×2160 pixel images at 30 frames/sec. An image compression module has been developed to improve the amount of video data that may be stored on our existing solid-state recorder. JPEG 2000 was chosen as the compression standard implementation for the following reasons:

1. Unlike earlier discrete-cosine-transform-based compression algorithms, which break the image down into small blocks, with JPEG 2000 the coding is global over the entire frame, thereby facilitating editing.
2. Truly lossless compression by a factor of 4 can be obtained when desired. Visually lossless compression is achievable with compression ratios ranging from 8 to 16. Compression ratios can be changed as a function of the application. Our present SSR, operating with state-of-the-art compact flash (CF) memory cards, can presently store approximately 8 minutes of video. A compression ratio of 10 will increase the recording time to 80 minutes.
3. The Digital Cinema Initiative (DCI) standard for ultra-high definition video is based on JPEG 2000 compression. The high-end resolution that DCI offers is that of our camera-recorder system. DCI is presently developing 3-D video standards that we will also endeavor to meet in the future.
4. Inexpensive integrated circuits already exist that allow us to provide a royalty-free compression function for of our camera component signals in real-time.

Various methods were examined to compress video in order to increase the record time of the SSR based upon three primary considerations: (1) ease of editing, (2) real-time recording, and (3) small size., combined with low power-consumption. MPEG compression relies on intra-frame coding that doesn't allow examination of single frames. This condition would not be acceptable for most Navy applications and has also been the

subject of many complaints in editing applications. We therefore limited our compression search to inter-frame coding schemes. Many compression processes have been developed to work on computers where time is not an issue; however, each of our four 1920×1080 camera output channels is generated at a 30 frame/sec rate – clearly unacceptable for our application. The most recent universal compression standard that has been introduced is JPEG 2000. This standard offers many benefits, including inter-frame coding, reasonably simple algorithms that can be implemented in hardware, and the possibility of operating in real-time. Interestingly, the JPEG 2000 standard incorporates many important psychophysics-based features that were investigated and developed in our Center in the early days video signal compression.

We initially considered developing a JPEG 2000 firmware algorithm but determined that such development would come at great cost. Reports from another R&D group describe real-time HDTV compression systems based subsystems developed by Vertex4 but considerable external memory is required. A firmware solution thus seemed out of the question. Alternative solutions were also investigated. For example, Mathstar developed a field programmable object array and demonstrated a JPEG 2000 decoder that worked. However the device was new, an encoder had not yet been programmed, and it would take learning time for a specialized part with a $50,000 licensing fee. Fortunately, commercial compression chips for HDTV signals became available in time for our use.

The Analog Devices ADV202 JPEG 2000 encoder/decoder was eventually selected as the integrated circuit to use for the compressor. This device provided all the features we needed:

1. No external parts or external support circuitry necessary.
2. Low power – 1.25 watts
3. Will compress 1920x1080 images at 30 frames/sec

The major disadvantage of the part was the poor documentation and reputation it received from other users. The overwhelming advantages of the part left us no other choice but to make it operate properly. Analog Devices improved upon the unit and created the pin-compatible ADV212 part that we are presently using.

The ADV212 is an extremely complex part, and we found the documentation to be poor and difficult to work with. In order to make the chip operate, a software program must be downloaded when the device first starts up to program it to be either an encoder or decoder, and dozens of hardware registers must be programmed in a precise order with handshaking. In order to gain experience doing this we designed a test module containing an ADV212 and targeted it to plug into an SSR motherboard with its CFs removed, thereby allowing it to act as a test base. Two test modules were inserted. One was programmed to be an encoder and the other a decoder. The SSR motherboard includes two SMPTE 292 HDTV serial ports. One was connected to one of the camera input channels, the other to a monitor output. This configuration allowed ADV212 development to continue with one ADV212 acting as an encoder and the second ADV212 acting as the decoder, which in turn drove the monitor. It took us quite a while, working closely with the Analog Devices technical staff, to fill in the missing documentation blanks in order to finally program the ADV212s and get them to function

as we desired. They presently operate properly, passing real-time video into one ADV212, compressing it using JPEG 2000 compression, passing the compressed video to the second chip programmed as a decompressor that restores the video and drives the serial output video stream that is in turn shown on a display.

During the period of development of the ADV212 software programming, we undertook a parallel effort to design a circuit board Compression Module (CM) containing four ADV212s, a Xilinx XC2V4000, double-data rates SDRAM, a Beck IPC PC on a chip with LAN access, and a CF memory card. The module is intended to accept the two SMPTE 292 serial camera channels, which contain a total of four HDTV images; separate the images sending an image to each ADV212; collect the output of each compressed image in the DDR-SDRAM; and then pass the combined set to the SSR.

This circuit board has been laid out, built and the components have been added. The device is presently undergoing testing to verify that everything is operational. Preliminary testing will be similar to the testing performed for the ADV212 test modules. Video will be input via one SMPTE 292 port passed to one ADV212 that is connected directly to a second ADV212. The output of the second will be passed to a SMPTE 292 output channel and then to a display. Once working, the other two ADV212s will be used in place of the first two.

A parallel firmware effort is also essentially completed. The goal is to accept the compressed output of the 4 ADV212s and collect them in a DDR-SDRAM. Once one compressed image is collected the memory will be available to collect subsequent images while the most recently available image is reformatted to appear as a 1920×1080 HDTV image that can be passed to the SSR for storage. Data will be transferred from the compression module to the SSR a LVDS transmission channel. The firmware requires independent processing of data received from each ADV212. The issue is compounded by the problem of the size of the compressed file varying from image to image for each UHD input image frame. Thus, image tracking relative to the UHD image becomes a significant concern. The DDR-SDRAM has the capacity of storing 8 complete compressed image frames to be stored before they need to be transferred to the SSR. The firmware has been structured with indexes so that ADV(0) can be storing its UHD(1) component, ADV(1) can be storing its UHD(2) component, ADV(2) can be storing its UHD(1) image component, and ADV(3) can be storing its UHD(3) image component, all while the composite JPEG 2000 image UHD(0) is being formatted and transferred to the SSR. This firmware has been simulated and works properly. An ADV212 data output firmware simulator is used to simulate all 4 ADVs on another SSR motherboard. Data is being collected and passed to the DDR-SDRAM. Data has been verified to be recovered from the memories. It will reformatted and passed via a high-speed serial channel to the SSR where it will be used in place of the SMPTE 292 inputs.

3.6. Projector Setup for 3D Viewing: Additional Details

The full impact of the 3D display that can be produced with a HDMAX camera pair is achieved only if the projected image is suitably large and the two component images are correctly aligned, requiring that they be of the same size and orientation to the horizontal. In the special monthly technical report for July 2007, titled *HDMAX Camera and Sony SRX-R105 Projector Configuration for 3D viewing: Typical Setup*, we discussed general

principles regarding the configuration of the HDMAX camera and projector pair. In this report we provide specific recommendations for the projector configuration.
The 3D projection system consists of the two SRX-R105 projectors and a 14′×7′ polarization-preserving projection screen. The projectors are large and heavy, measuring approximately 53 inches in length by 30 inches in width by 18 inches in height and weighing 250 lbs each. Indeed, part of the challenge of setting up the projection system is simply dealing with the bulk of the projectors. It should be noted that the Sony projectors provide the only display capability for the cameras at this time. They must thus be set up and operating in order for the full ultra-high-definition camera output(s) to be seen. Two additional Sony-supplied documents are sent with this report. One is a full-color advertising brochure describing the projector system, the other is the operation manual. Figure 5 provides some sense of our own camera/screen installation. The room—large but otherwise standard (i.e., not configured as a theater)—is 40 feet long by 9 feet high. The screen fills most of the wall at one end of the room. One problem with this arrangement is the shadowing of the screen by anyone sitting closer than about 15 feet from the screen. Since the optimum viewing distance is between 10 and 12 feet for a 7′-high HDMAX image, some minor vision-produced loss in image resolution is experienced.

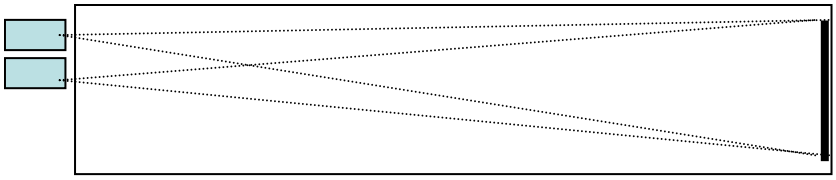


Figure 3.5. Vertically-stacked projector configuration. The projectors produce overlapping images on a 14′×7′ polarization-preserving screen some 40′ distant. Note that with this configuration, it is difficult to avoid shadows of the heads of viewers who sit the optimum 10-12′ from the screen.

The following information and guidelines will aid in the setup of a suitable viewing "studio."

1. The projection screen is 100 inches high by 170 inches wide, including a 3-inch frame/border around all sides. Ideally it should be mounted at such a height that it can be viewed without excessive head tilt, either up or down, from a viewing distance of 10-12 feet. In our case, this meant mounting the screen as high as possible in the room (in fact, given the height of the ceiling, "as high as possible" was only a short distance higher than "as low as possible"). If possible, the projectors and screen should in fact be installed in a small theater, with people in the middle row of the theater viewing the screen approximately head-on and from the 10-12-foot viewing distance but producing no shadows on the screen.
2. The projectors require single-phase 220-volt power at 50-60 Hz. Each projector draws nearly 14 amps when operating (~3 kW of power).
3. The projectors should be level, one projector above the other. If the room is of the nontheater configuration, as is ours, it is important that the projectors be positioned as high as possible to minimize the shadowing problem noted above. With proper leveling of the projectors, (a) their optical axes will be perpendicular to the projection screen, as desired, and (b) the two images will be square with

one another, i.e., with no relative rotation of one with respect to the other. During this alignment/registration operation, the projectors should be instructed to project one of the two alignment test patterns on the screen. One test pattern consists of a blackwhite (or, in reversed mode, white-black) checkboard. The other test pattern consists of a grid of fine, green, horizontal and vertical lines. Two people are typically required for the aligment procedure, one viewing the image on the screen from a short distance, the other adjusting the projectors for relative tilt.

4. The two images can be brought into registration by means of the motor-actuated zoom and vertical-position controls. The vertical positioning control automatically corrects for keystoning of the images, maintaining their "squareness." It is not essential that the two images be perfectly registered—there can, and probably will, be some slight misregistration at the edges of the images. However, the images should be adjusted to be as well-registered as possible. Horizontal misregistration will manifest itself as distortions of the scene in the depth direction. A small amount of vertical misregistration, including misregistration in angular orientation, can be tolerated by the viewer, but it should not be allowed to be too large: an inch of vertical misregistration on the screen is close to the maximum that we have allowed in our own projections.
   Note that image keystone correction is the reason the two projectors must be positioned one above the other rather than side-by-side. Were they mounted side-by-side, problems with head shadows on the screen would be much less severe, but the uncorrectable keystoning would produce faulty 3D images.
5. Once the projected images are in suitable alignment, the polarizing sheets can be positioned, one in front of each projector. We used polarizing sheets that polarize the light from the projectors at ±45° to the vertical, matched by viewing glasses polarized the same way. (The polarization axis of a linear polarizer sheet can be determined by viewing a reflection of a light off a polished non-metallic surface at a viewing angle of around 45°: The reflection will be brightest if the polarization axis of the polarizer is horizontal.)

4.0 STEREO AND MULTI-VIEW IMAGE AND VIDEO CODING, TRACKING, ANALYSIS AND PLAYBACK

PI: Dr. Borko Furht

Tasks 3.28-3.30

4.1 Summary

This section provides technical documentation of the third year of research activities in the field of image and video analysis algorithms for coastline security. Our work in the third year has been focused on developing robust techniques and methodologies for multi-view video investigating and developing techniques, technologies, and algorithms needed to create and analyze 3D images and 3D videos provided by multiple high-definition cameras with the specific focus on coastline security applications. This works extends our efforts from the first two years. As a set of deliverables of the second year research we proposed and implemented algorithms for multi-view video compression as well as tracking of video objects and video analysis using depth information. Figure 4.1.1 illustrates how these research efforts fit in the overall surveillance system based on multiple mounted high-definition cameras.

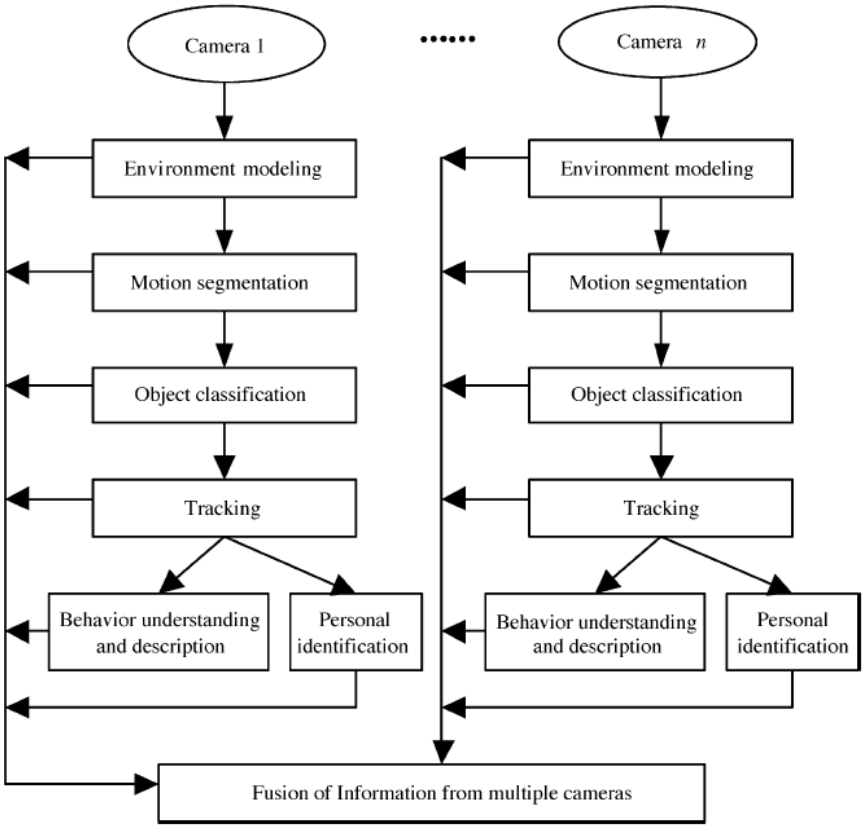


Figure 4.1.1. General framework of a multi-camera video surveillance system

4.2 Introduction

4.2.1 Project Description

This report describes the work involved in the development of a video surveillance system for the Center of Coastline Security Technology. The ultimate goal of our work is to provide semi-automatic tools to monitor marine traffic at key locations, analyzing the contents of incoming video streams, detecting potential threats, and triggering the corresponding action.

Our efforts during Year 3 of the grant have been focused mostly on merging algorithms and techniques developed in the last two years of this project into a robust multi-view video surveillance system applicable to maritime domain. The developed methods for such automated surveillance are ultimately targeted for real-time or near-real-time processing from sequences obtained by both regular cameras as well as high-definition cameras supporting HDTV and/or QuadHDTV resolutions.

4.2.2 Project Scope and Objectives

This project is part of the Center of Coastline Security Technology at Florida Atlantic University. It is expected to be integrated at the output of the video capture stage developed by Dr. Bill Glenn's group. The objectives for Year 3 are:

- *Develop effective methods to 3D video compression, delivery and playback.* Techniques and methods for efficient compression of stereo and multi-view sequences is an ongoing research area. It is anticipated that 3D video improves surveillance applications. 3D autostereoscopic displays (no glasses required) are recently being released and are becoming notably inexpensive. The goal is also to create a 3D video player for Sharp autostereoscopic display, which is one of the first commercially available autostereoscopic displays.
- *Develop purely computational as well as biologically plausible methods and algorithms for detection, tracking and classification of video objects using depth information acquired from multiple cameras.* In addition to investigating classical, purely computational models for detection of video objects, we also studied models inspired by principles of human visual attention. While a single-view object segmentation is limited in the sense that the occlusion is difficult to detect and it is difficult to distinguish far objects from close ones, segmentation with depth information allows for easy occlusion detection, helps distinguishing far from close objects, and helps the task of classification and tracking. Depth information provides an important feature of detected video objects that can be used for further analysis.

4.2.3 Project Team

Faculty:
Dr. Borko Furht, PI
Dr. Taghi M. Khoshgoftaar, co-PI
Dr. Oge Marques
Dr. Hari Kalva
Dr. Daniel Socek

Students:
Lakis Cristodoulou, Alvaro Fonseca, Qiming Luo, Liam Mayron, Carlos Pertuz, and Xiaoyuan Su

4.3 Multiple Object Tracking System for Traffic Surveillance and a Progressive Edge-Based Stereo Correspondence Method

Section 4.3 of this report describes two sub-projects: (1) a practical rule-based multiple object tracking system for traffic surveillance, and (2) a progressive edge-based stereo correspondence method.

4.3.1 A Practical Rule-Based Multiple Object Tracking System for Traffic Surveillance

We propose a novel and effective rule-based multiple object tracking system for traffic surveillance using a collaborative background extraction algorithm, which collaboratively extracts a background from multiple independent extractions to remove spurious background pixels. The multiple object tracking is based on differenced binary images between video frames and the extracted backgrounds and is therefore simplified. The rule-based strategies are applied for thresholding, outlier removal, object consolidation, neighboring objects separation, and shadow removal. Empirical results show that our multiple object tracking system is highly accurate for traffic surveillance under conditions of occlusion and background variations.

4.3.1.1 Introduction and Related Work

Multiple object tracking (*MOT*) is important for visual surveillance and event classification tasks [1]. However, due to challenges such as background variation, occlusion, and object appearance variation, *MOT* is generally difficult. In the case of traffic surveillance, *background variations* in terms of illumination variation, small motions in the environment, weather and shadow changes, *occlusions* in terms of vehicles overshadowed or blocked by neighboring vehicles, trees, or constructions, and *vehicle appearance changes* in terms of different sizes of the same vehicles in different video frames, contribute to inaccurate visual tracking.

Traditional visual tracking methods include feature-based tracking, template tracking, Kalman filtering [2], and kernel particle filtering [3][4]. Feature-based tracking detects features in a video frame and searches for the same features nearby in subsequent frames. Template tracking tracks a fixed template through a sequence of images. However, these two tracking methods are computationally inefficient. Kalman filtering uses a linear function of parameters with respect of time, and assumes white noise with a Gaussian distribution; however, the method with the Kalman filtering to predict states of objects can not be applied to objects in occlusion [5]. Particle filtering is appealing in *MOT* for its ability to have multiple hypotheses; however, direct application of particle filtering for multiple object tracking is not feasible [3].

In this paper, we proposed a rule-based multiple object tracking system using a collaborative background extraction algorithm for the application of traffic surveillance. This system is easy-to-implement as it is based on the binary images from thresholding the video frames with the backgrounds, and it is highly effective in handling occlusions in terms of removing outliers and shadows, consolidating objects, and separating occluded vehicles. The collaborative background extraction algorithm collaboratively extracts a background from several independent extractions of the background, which effectively removes spurious background pixels and adaptively reflects the environment changes.

Section 4.3.1.2 introduces related work. Section 4.3.1.3 is our framework for the collaborative background extraction algorithm and rule-based multiple object tracking system for traffic surveillance. Experimental results and conclusions are in Section 4.3.1.4 and Section 4.3.1.5.

4.3.1.2.1 Background Extraction

For traffic surveillance videos that generally have stationary background, it is important to segment the moving vehicles from the background either when viewing the scene from a fixed camera or after stabilization of the camera motion. With the assumption of a stationary camera, we can simply threshold the difference of intensities between the current video frame with the background image, $I(x,y)-I_{bg}(x,y)$, to segment the moving objects from the background. However, due to background variations, this simple approach may not work well in general.

As each pixel in the background varies in a different way over time, a normal (Gaussian) distribution $N(\mu,\sigma)$ can be used to model the changes. When we have a set of images, the background can be modeled by computing the mean value μ and standard deviation σ on a pixel-by-pixel basis. A pixel in a new image can be classified as a *background* pixel if $|I(x,y)-\mu|<k\sigma$, otherwise as *foreground*, where k is a confidence parameter with k=3 producing 99% confidence. An improved Gaussian model, called a *mixture model*, uses a weighted sum of normal distributions to classify *background* and *foreground*. The probability for a *background* pixel to have value v can be modeled as:

$$P(I(x,y)=v)=\sum_i w_i e^{-\frac{(v-\mu_i)^2}{2\sigma_i^2}} \quad (1)$$

where weights w_i, means μ_i, and standard deviations σ_i are learned from training images and can be updated over time to reflect gradual changes of environment [6].

Linear prediction and adaptation can be used to model background changes over time, for example

$$\mu_t=(1-\rho)\mu_{t-1}+\rho X_t \quad (2)$$

where μ_t is the mean value at time t, ρ is a learning rate, and X_t is the intensity value of the new pixel [6].

Hysteresis thresholding [7] can be used to segment the background in a way favoring connected regions instead of individual pixels. A conservative threshold T_h is used to classify *foreground* objects with high confidence, and the pixels connected with

foreground objects that satisfy a less conservative threshold T_l are also classified as *foreground*.

4.3.1.2.2 Multiple Object Tracking

Multiple object tracking is known as a difficult research topic in computer vision. It has to deal with the difficulties for single object tracking, such as changing appearances, non-rigid motion, dynamic illumination and occlusion, as well as the problems related to multiple object tracking including inter-object occlusion, multiple object confusion, etc.

Traditional visual tracking methods for single object tracking, such as Kalman filtering [2], hidden Markov models [8], and kernel particle filtering [3][4], can handle noise, occlusions, and perform some form of error correction. Multiple object tracking requires different approaches when the objects are described using the same models. Instantiating independent trackers for each object is not an ideal solution because the independent trackers tend to join together onto a single object.

There has been much work on multiple object tracking. *Intille et al.* interpreted the targets as blobs which merge and split [9]; *Rasmussen and Hager* enforced a minimum separation between targets [10]; *Isard and MacCormick* propose a Bayesian multiple-blob tracker [11]; *Hue et al.* proposed an extension of classical particle filter where the stochastic assignment vector is estimated by a Gibbs sampler [12]; the Probabilistic Data Association Filter (*PDAF*) [13][14] extended the Kalman filter [2] by using a Bayesian approach to the problem of data association (how to update the state when there is a single target and possibly no measurements or multiple measurements due to noise); the Joint Probabilistic Data Association Filter (*JPDAF*) [14][15] enforces a kind of exclusion principle that prevents two or more trackers from latching onto the same target by jointly calculating target-measurement association probabilities. Reid and Murray [16] developed an active tracking system using affine structure, which can learn about its target and use the knowledge to improve its performance.

For applications in traffic tracking and human being tracking, *Bai et al.* proposed feature-based and appearance-based approaches for traffic tracking [17]. *Beleznai et al*. [18] implemented a fast mean shift model to track human beings and achieved better performance than blob-based approaches [19][20].

4.3.1.3 Framework

Our multiple object tracking system has the following procedure: adaptively extract backgrounds using collaborative background extraction, generate binary images by differencing video frames with their backgrounds, apply the rule-based tracking strategies including outlier removal, object consolidation, neighboring objects separation, and shadow removal, and finally record features of tracked objects.

4.3.1.3.1 Collaborative Background Extraction

We propose a non-Gaussian, single thresholding background extraction method called collaborative background extraction, an adaptive background extraction algorithm using the idea of collaborative filtering [21].

With the assumption that the background will not change significantly in a few seconds, we extract several backgrounds alternatively over a short period of time, e.g., every 60 frames or every 2 seconds for 30 fps videos, and then integrate these backgrounds into one. By updating the background every few seconds, we adaptively model the background changes.

As illustrated in Fig. 4.3.1.1, every single background extraction will produce a background with spurious points in different locations (Fig. 4.3.1.1(a)~(d), black points in the first four background images are spurious background points). With the help of collaborative extraction, the final background (Fig. 4.3.1.1(e)) is almost impeccable.

Collaborative filtering is a technique that has been successfully applied in recommendation systems, in which recommenders may collaboratively make recommendations for recipients on interesting items, in addition to indicating those that should be filtered out [21].

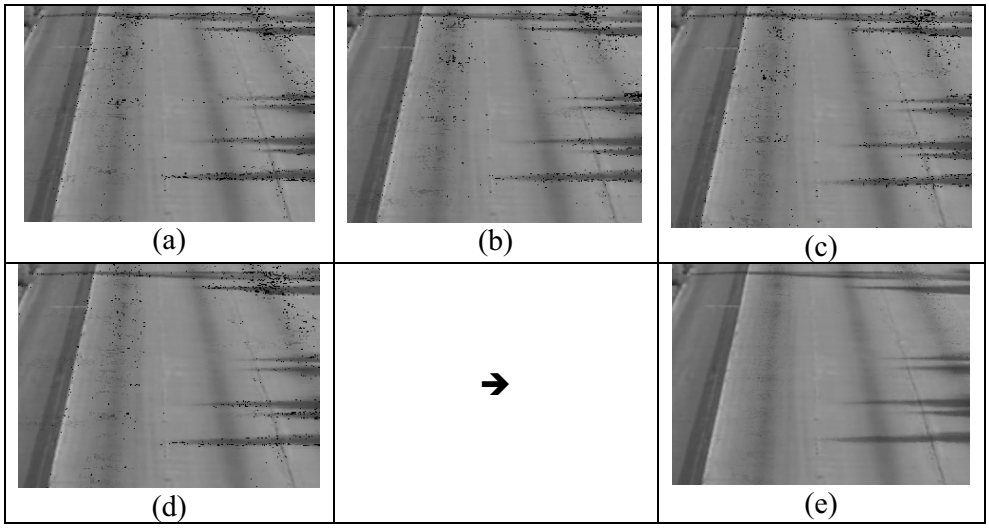


Figure 4.3.1.1. Collaborative background extraction for a traffic surveillance video. (a) background extracted from frame 1, 5, 9, … 57; (b) background extracted from frame 2, 6, 10, … 58; (c) background extracted from frame 3, 7, 11, … 59; (d) background extracted from frame 4, 8, 12, … 60; (e) the final background resulted from collaboratively filtering the multiple background extractions, with spurious background pixels removed.

We borrow the idea of collaborative filtering to collaboratively extract a background from four independent extractions of background, in order to produce a

reliable background from multiple individual extractions. However, we do not classify a *foreground* pixel when there are more *foreground* labels than *background* ones out of the 4 single background extractions. We use the average intensity value of the labeled *background* pixels instead.

For each individual background extraction, we use a small threshold of intensity difference (we use 2 here, according to our empirical experiments) to determine *background* and *foreground*, and assign 0 for a *foreground* pixel and an average intensity value for the *background* pixel (Equation 3). We do not threshold immediately consecutive frames; instead, we calculate an intensity difference between frames with a gap of four frames. This strategy is specifically for the traffic videos, in which vehicles have generally moved out of the locations of four frames before as they move very fast.

$$bg_{k,i}(a,b)=\begin{cases}\dfrac{1}{|S|}\displaystyle\sum_{s\in S}\dfrac{I_s(a,b)+I_{s+4}(a,b)}{2}, & S=\{s\,\|\,I_s(a,b)-I_{s+4}(a,b)|\le 2\},|S|\neq 0\\ 0, & |S|=0\end{cases}$$

(3)

where k is the index number of the starting frame of background extraction, i is the index number of the four independent background extractions, $s\in[k+i-1 : 4 : k+i+51]$, which means s is between $k+i-1$ and $k+i+51$, with an increment of 4 in each iteration, and 51 is a value calculated from a total consecutive frame number of 60 (which can be adjusted for actual videos) for each individual background extraction. E.g., given $k=1$, $i=4$, we will have $s\in\{4, 8, ..., 56\}$, and we need to calculate $|I_{56}(a,b)-I_{60}(a,b)|$ for $s=56$. S is the set containing the frame indices of any pixels at (a,b) such that the intensity difference $|I_s(a,b)-I_{s+4}(a,b)| \le 2$. During the iterations of $[k+i-1 : 4 : k+i+51]$ (from k+i-1 to k+i+51, with an increase of 4 in each iteration), when $|I_s(a,b)-I_{s+4}(a,b)|\le 2$, we take the average intensity value of frame s and $s+4$, $I_{ave}(s,s+4)=[I_s(a,b)+I_{s+4}(a,b)]/2$. For example, when we have three occasions that $|I_s(a,b)-I_{s+4}(a,b)|\le 2$ for $k=1$, $i=1$, at $s=21$, 33, and 53, we will have $bg_{1,1}(a,b)=[I_{ave}(21, 25)+ I_{ave}(33, 37)+ I_{ave}(53, 57)]/3$. If none of the intensity differences is less than or equal to 2, we will have $bg_{1,1}(a,b)=0$.

The four independent background extractions are produced by alternatively thresholding the frames, e.g., the 1st individual background is extracted from frames *1, 5, 9……, 57,* the 2nd from frames *2, 6, 10, ……, 58,* the 3rd from frames *3, 7, 11, ……, 59,* and the 4th from frames *4, 8, 12, ……, 60* (Fig. 4.3.1.1).

After the 4 independent backgrounds have been extracted, we produce the final background by using the average intensity value of the labeled *background* pixels from the 4 extractions, and those *foreground* pixels (with values of 0) are then automatically replaced unless none of the four is classified as *background* (Equation 4).

$$bg_k(a,b)=\begin{cases}\dfrac{1}{|I|}\displaystyle\sum_{i\in I} bg_{k,i}(a,b), & I=\{i\,|\,bg_{k,i}(a,b)\neq 0; i=1,2,3,4\},|I|\neq 0\\ 0, & |I|=0\end{cases}$$

(4)

where k is the index number of the starting frame of background extraction, i is the index number of the four independent background extractions, I is for a non-zero pixel at (a,b) of the four background files. For example, for a pixel (a,b), when given $bg_{k,1}(a,b)=0$, $bg_{k,2}(a,b)=0$, $bg_{k,3}(a,b)=20$, and $bg_{k,4}(a,b)=30$, we will have $|I|=2$, and $bg_k(a,b)= (20+30)/2=25$.

The detailed collaborative background extraction algorithm is described in Fig. 4.3.1.2.

Algorithm: Collaborative Background Extraction
{ **Input**:
k = starting frame index;
total_fm = total frame number of the video;
FN = 60 (total consecutive frame numbers for each independent background extraction);
Th = 2 (threshold);

Output:
extracted background files bg_k (for each of the starting frame numbers k).
};

begin
For (k=1; k<total_fm; k=k+FN)
{ Initialize each pixel (a,b) in $bg_{k,1}$, $bg_{k,2}$, $bg_{k,3}$, $bg_{k,4}$ to 0

Call functions:
$bg_{k,1}$=bg_extra(k, FN-8, Th);
$bg_{k,2}$=bg_extra(k+1, FN-8, Th);
$bg_{k,3}$=bg_extra(k+2, FN-8, Th);
$bg_{k,4}$=bg_extra(k+3, FN-8, Th);

Calculate $bg_k(a,b)$=ave_bg($bg_{k,1}$, $bg_{k,2}$, $bg_{k,3}$, $bg_{k,4}$) for each pixel (a,b) of the background file using Equation 4;
}
end.

Function: Independent background extraction (bg_extra)
{ **Input**: starting_frame_index sf,
extraction_frame_nums fn,
threshold Thres
Output: background_file bg_i
};

begin
Initialize num(a,b)=0 and bg_f(a,b)=0 for each pixel (a,b)
for (x=sf; x<sf+fn; x=x+4)
{ h1=frame(x);
h2=frame(x+4);
diff=abs(h2-h1);
for each pixel in the frame
{ if (diff(a,b)<=Thres)
{ num(a,b)=num(a,b)+1;
bg_f(a,b)=
bg_f(a,b)+(h1(a,b)+h2(a,b))/2;
}
}

```
}

for each pixel (a,b) in the background file
{  if (num(a,b)≠0)
     bg_i(a,b)=bg_f(a,b)/num(a,b);
   else
     bg_i(a,b)=0;
}
end.
```

Figure 4.3.1.2. Collaborative background extraction algorithm.

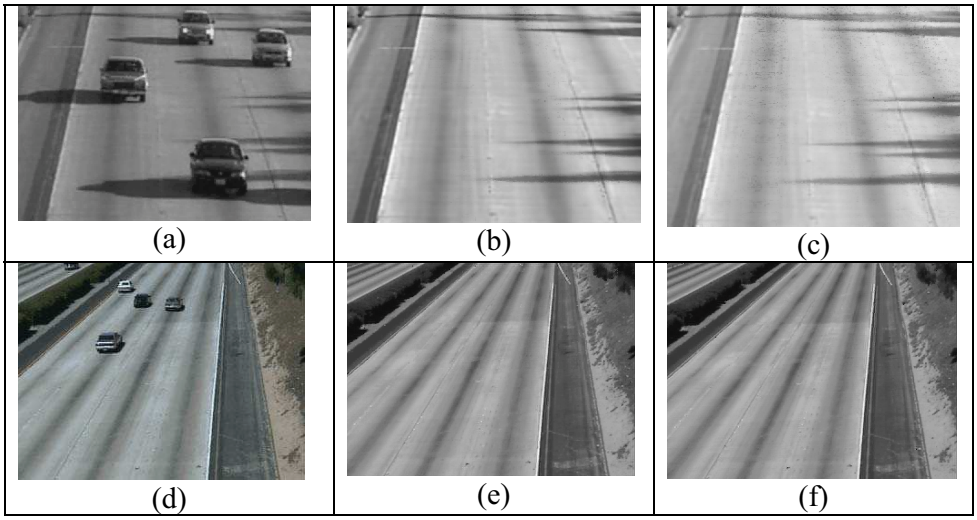


Figure 4.3.1.3. Collaborative background extraction algorithm for traffic surveillance (left column, on traffic video clip I) (a) frame 84 , (b)background extracted from frames 61~120 (c) background extracted from frames 301~360; (right column, on traffic video clip II) (d) frame 350 (e) background extracted from frames 61~120, (f) background extracted from frames 301~360.

In our implementation, we adaptively extract backgrounds every 60 frames. However, this parameter is adjustable according to actual video clips.

Fig. 4.3.1.3 illustrates the results of our adaptive background extraction. The two backgrounds in Fig. 4.3.1.3(b) and (c) were extracted from frames 61~120 and 301~360 of our traffic surveillance Video I (see details about the videos in Section 4.3.1.4, Table 4.3.1.1). Although the time difference is just a few seconds, we find there are apparent changes between these two backgrounds, e.g., the brightness of the backgrounds and the lengths of shadows are different. For video clip II, however, the background does not change much over time. Fig. 4.3.1.3 (e) and (f) are backgrounds extracted also from frames 61~120 and frames 301~360, which remain almost the same.

4.3.1.3.2 Rule-based Multiple Object Tracking for Traffic Surveillance

After adaptively extracting backgrounds, our multiple object tracking for traffic surveillance can be simplified and be based on the binary images from thresholding

the video frames with the backgrounds. The rule-based steps are the following (see Fig. 4.3.1.4): (1) difference each video frame with its background to produce a binary image, (2) clean up the binary image and consolidate objects using outlier remover, hole remover, and strip removers, (3) separate neighboring vehicles and remove shadows, (4) clean up the binary image again using outlier strip and hole strip removals. Finally we extract and record the features of tracked vehicles.

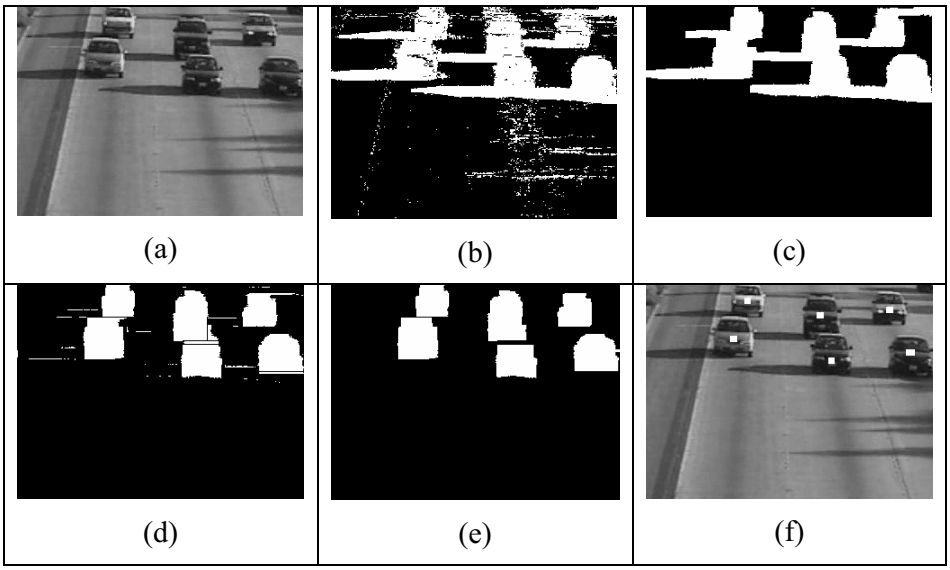


Figure 4.3.1.4. Steps of rule-based multiple object tracking system for traffic surveillance (a) frame No. 139 of highway traffic Video I, (b) after differencing the video frame with its background, (c) after outlier removal and hole removal, (d) after object separation and shadow removals, (e) after outlier strip and hole strip removals again, (f) tracked multiple vehicles (with white squares in the geometric centers of the vehicles).

4.3.1.3.2.1 Differencing with Background

From input frame I_t at time t, we can create a binary image I_b by differencing its intensity value with that of the background image I_{bg}. We use a threshold value of 10 (according to our empirical experiments on the traffic videos I and II), i.e., when $I_t(a,b)-I_{bg}(a,b)>10$, $I_b(a,b)=1$; else, $I_b(a,b)=0$.

4.3.1.3.2.2 Outlier Removal

As our objects to track are vehicles, they are supposed to have connected blocks with solid areas (with non-negligible heights and widths) in their respective binary images. While unconnected blobs that are much smaller than a car are considered outliers and need to be cleaned up. Fig. 4.3.1.5 is the illustration of outlier removal and hole removal. Our outlier removal algorithm is in Fig. 4.3.1.6.

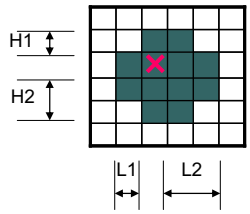


Figure 4.3.1.5. Illustration for outlier removal and hole removal: Outlier removal: If H1+H2+L1+L2<threshold, X (foreground pixel) will be replaced by a background pixel; Vice versa for hole removal.

Algorithm: Outlier removal algorithm
{ **Input**:
in_f = input binary image
thres = a threshold for outlier removal;
Output:
binary image with outliers removed.
};

begin
for each pixel (a,b) in the binary image
{ if(in_f(a, b)==1)
{search in four directions (up, down, left, right) until the pixel is 0 in each direction
compute the total distance to reach 0s:
total_dis=up_dis+down_dis+left_dis+right_dis;
if (total_dis<=thres)
{
in_f(a,b)=0;
}
}
}
end.

Figure 4.3.1.6. Outlier removal algorithm.

4.3.1.3.2.3 Hole Removal (Object Consolidation)

When some portions of a vehicle have similar intensity values to the background, these portions will be left as holes in the binary image after the intensity differencing. The removal of the holes will connect the unnecessarily separated regions and consolidate the tracked objects. The hole removal algorithm is in Fig. 4.3.1.7 (an illustration figure is shown in Fig. 4.3.1.5).

Algorithm: hole removal algorithm
{ **Input**:
in_f = input binary image

```
   thres = a threshold for hole removal;
  Output:
   binary image with holes removed.
};

 begin
for each pixel (a,b) in the binary image
 {
  if(in_f(a, b)==0)
    {search in four directions (up, down, left, right) until the pixel is 1 in each
    direction
   compute the total distance to reach 1s:
     total_dis=up_dis+down_dis+left_dis+right_dis;
   if (total_dis<=thres)
   {
     in_f(a,b)=1;
   }
  }
}
end.
```

Figure 4.3.1.7. Hole removal algorithm.

4.3.1.3.2.4 Strip Removal

For strip-shaped outliers or holes that a regular outlier remover or a hole remover with small thresholds can not remove (increasing the threshold may result in over-pruning), we use strip removers for outliers and holes instead. The basic idea of outlier strip removal is that if the pixels at the four corners of the rectangle (strip) with height *h* and width *w* and centered at the pixel (a,b) are all 0s (all 1s for hole strip removal), then each pixel in the rectangle will be set to 0 (set to 1 for hole strip removal). Fig. 4.3.1.8 is an illustration figure for both outlier strip removal and hole strip removal algorithm. Fig. 4.3.1.9 is our outlier strip removal algorithm, and Fig. 4.3.1.10 is the hole strip removal algorithm. When we want to remove a horizontal strip, we can set the height to a certain value and set the width as 1 (here the rectangle is reduced to a line); while for vertical strip, we can set the height as 1 and the width a certain value; if each of the width and the height is bigger than 1, the strip remover will remove both horizontal and vertical strips. The setting of the width and the height is rule-based and adjustable.

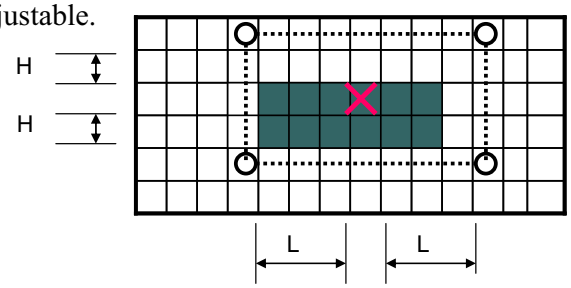


Figure 4.3.1.8. Illustration for outlier strip removal and hole strip removal: Outlier strip removal: If four corner pixels of a rectangle centered at X (foreground pixel) are

all background pixels, all pixels within the rectangle will be replaced by background pixels; Vice versa for hole strip removal.

```
Algorithm: Outlier strip removal algorithm
{ Input:
  in_f = input binary image
  h = height of a rectangle;
  w = width of the rectangle;
 Output:
    binary image with outlier strips removed.
};

 begin
for each pixel (a,b) in the binary image
 {
 if(in_f(a,b)==1)
 {
    if the pixels at the four corners of the rectangle with h*w and centered at (a,b)
      are all 0s
   {
    for each pixel (i,j) in the rectangle
    {
     in_f(i, j)=0;
    }
   }
 }
}
 end.
```

Figure 4.3.1.9. Outlier strip removal algorithm.

```
Algorithm: Hole strip removal algorithm
{ Input:
  in_f = input binary image
  h = height of a rectangle;
  w = width of the rectangle;
 Output:
  binary image with hole strips removed.
};

 begin
for each pixel (a,b) in the binary image
 {
 if(in_f(a,b)==0)
 {
   if the pixels at the four corners of the rectangle with h*w and centered at (a,b)
   are all 1s
   {
    for each pixel (i,j) in the rectangle
    {
     in_f(i, j)=1;
```

```
    }
  }
}
}
end.
```

Figure 4.3.1.10. Hole strip removal algorithm.

4.3.1.3.2.5 Shadow Removal

As our task is to track vehicles, not their shadows, we need to effectively remove the shadows in our tracking system, especially when shadows are longer than the widths of vehicles. In the example of Fig. 4.3.1.4, we first need to locate where to start the shadow removal (e.g., the vertical rods together with arrows in Fig. 4.3.1.11), we also need to create vertical background strips to protect the vehicles from mis-pruning in the binary image (e.g., the vertical rods the arrows are pointing to in Fig. 4.3.1.11), and then we can remove the shadows.

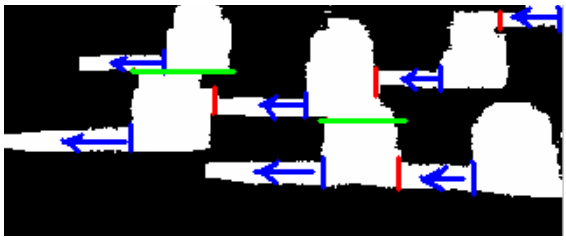
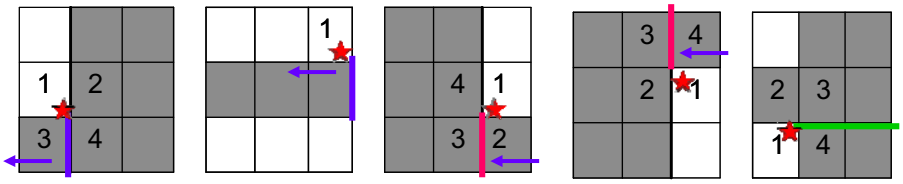


Figure 4.3.1.11. Separations of neighboring vehicles (green horizontal lines) and locations of where to start removing shadows (blue vertical rods together with arrows) and where to stop (red vertical rods the arrows pointing to).



(a) (b) (c) (d) (e)

Figure 4.3.1.12. Corner locations for shadow removal starts, removal protection stops, and vehicle separations (the corners are around the red stars in the square 1 of each figure, white cells are background and gray cells are objects) (a) inner corner for start of removing leftwards shadows (from vehicle), (b) inner corner for starts of removing leftwards shadows (from the right border), (c)(d) inner corners where leftwards shadow removal protection stops, (e) inner corner for vehicle separation.

Locating corners of where to start removing shadows and where to stop is rule-based: we need to detect four kinds of inner corners illustrated in Fig. 4.3.1.12. (a) is the corner for the start of leftwards shadow removals, (b) is the corner for the start of leftwards shadow removals outside of the images (on the right border), and (c)(d) are the corners for removal protection of leftwards shadow pruning.

Except for Fig. 4.3.1.12(b), each of the inner corners meets the following rules: (1) starting from the corner, the height of the shadow is smaller than the width or height of the vehicle, (2) for leftwards shadows, the corners of the starting points and stop points for shadow removal have three kinds of shapes illustrated in Fig. 4.3.1.12 (a)(c)(d), (3) the square centered at the corner can be approximately composed by one *background* square (0s, e.g., square 1 in Fig. 4.3.1.12(a), black pixels in Fig. 4.3.1.11) and three *foreground* squares (1s, e.g., squares 2, 3, 4 in Fig. 4.3.1.12(a), white pixels in Fig. 4.3.1.11). For the 3rd rule, we relax the constraint to allow bigger flexibility: the number of *background* pixels in square 1 of Fig. 4.3.1.12 is 22%~28% (around 1/4) of total pixels of the square covering the 4 small squares 1~4 of Fig. 4.3.1.12, and the pixel number of *foreground* squares in other three squares is 72%~78% (around 3/4) of the total. The size of the squares is rule-based and adjustable for different videos.

The rule to locate the inner corner for leftwards shadow removal starting from the right border (Fig. 4.3.1.12(b)) is: near the border, the ratio of height to width of the object in the binary image is less than 0.4 (this ratio is adjustable according to actual videos).

After locating the shadow corners, we start removing the shadows leftwards until the shadows are removed or we meet removal protection stops (e.g., Fig. 4.3.1.12(c)(d)), which are vertical background strips created to separate the shadow from the vehicles.

The above rule-based strategies are for the leftwards shadow removals. When the traffic surveillance camera is horizontally placed and is vertical to the traffic, the shadows have three situations, leftwards shadows, rightwards shadows, and short or no shadows. The last case of shadows is generally not considered for shadow removing. For rightwards shadows, the starts and stops in the shadow removal are reversed with those for leftwards shadows.

4.3.1.3.2.6 Object Separation

When shadows are presented in traffic surveillance videos, we separate the partially occluded vehicles from their neighbors in the binary image by creating narrow horizontal *background* strips (0s), with the help of shadows. We only consider one shape of corners for leftwards shadows to separate connected vehicles in the binary image (see Fig. 4.3.1.12(e)): the bottom of one vehicle (or its shadow) is neighboring the top of another vehicle in a traffic image (also see the example in Fig. 4.3.1.11). The rule to determine the corner is similar to that for shadow removal: it requires 22%~28% (around 1/4) 0s in the *background* square and 72%~78% (around 3/4) 1s in the *foreground* squares, in addition to having the corner shape of Fig. 4.3.1.12(e). The objects separation is conducted before shadow removals.

4.3.1.3.2.7 Feature Recording

After applying the above rule-based multiple object tracking strategies, it will be easy for us to record the heights, widths, and geometric centers for each tracked vehicle. These features in consecutive video frames are used for further tracking tasks, such as track analysis, event classification, and abnormal behavior alarming. For example, in Fig. 4.3.1.13, through recording the geometric centers of tracked vehicles, we can

easily classify the traffic event of lane changing of vehicles, despite occasional false tracking of the vehicle centers.

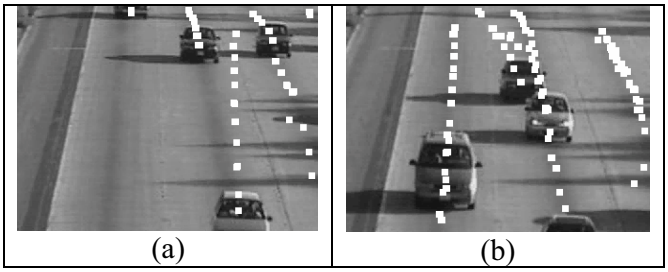


Figure 4.3.1.13. Lane changing in the traffic video clip I, (a) recorded vehicle tracks of frames 65~75, (b) recorded vehicle tracks of frames 320~332.

4.3.1.4 Experimental Results

We implement our collaborative background extraction algorithm and rule-based multiple object tracking strategies for traffic surveillance, and experiment on two highway traffic videos, one with heavy occlusions from shadows and frequently changing backgrounds (Video I), another with barely any shadows and an almost fixed background (Video II) (see frame examples of the two video clips in Fig. 4.3.1.3(a) and Fig. 4.3.1.3(d)).

We adaptively extracted backgrounds using the collaborative background extraction algorithm for both videos. Then, we applied all the steps of rule-based multiple object tracking for Video I, and used only the basic steps of background differencing, outlier removal, and object consolidation for Video II.

The summary of the videos and the experimental results are in Table 4.3.1.1. We use hit rate (the rate of the correctly tracked objects by the total number of objects, see white squares within the vehicles in Fig. 4.3.1.14) and false alarm rate (the rate of the incorrectly labeled objects by the total number of objects, see white squares outside the vehicles in Fig. 4.3.1.14) as the performance metrics.

Table 4.3.1.1. Video information and tracking results of a heavily-occluded video (Video I) and an occlusion-free video (Video II).

Videos	I (heavily-occluded video)	II (occlusion-free video)
number of frames	440	500
time duration	29"	33"
frame size	240*320	240*320
valid vehicles	1684	2088
correctly tracked vehicles	1649	1997
hit rate	97.92%	95.64%
false alarm rate	1.37%	0

Experimental results show that the hit rate of our rule-based multiple object tracking system for the heavily-occluded video (Video I) is higher than the occlusion-free video (Video II), and the false alarm rate for Video I is very low. The reason that the Video II has a zero false alarm is because it is occlusion-free and nothing else can be mistaken as vehicles. With the high effectiveness of our rule-based strategies, we achieve a very accurate tracking performance in terms of a very low false alarm rate and a higher hit rate on the heavily-occluded video than the occlusion-free video that does not apply our occlusion handling methods.

Fig. 4.3.1.14 illustrates the situations where our tracking system produces false alarms. The false alarms in Fig. 4.3.1.14(a) and Fig. 4.3.1.14(c) are caused by external shadows from the right border that are not effectively removed by the rule-based tracking system, the false alarms in Fig. 4.3.1.14(b) are caused by the shadow from an exceptionally big vehicle, and a shadow of an external vehicle from the right border; however, in each of the neighboring frames, our tracking system has much better tracking accuracy in terms of increased correctly tracked vehicles and decreased incorrectly tracked vehicles (Fig. 4.3.1.14(d)(e)(f)). These examples indicate that our rule-based tracking system can produce inaccurate tracking where the rules do not apply; however, tracking accuracy can be improved and false alarms can be eliminated by adjusting the comprehensive rules customized for different surveillance environments.

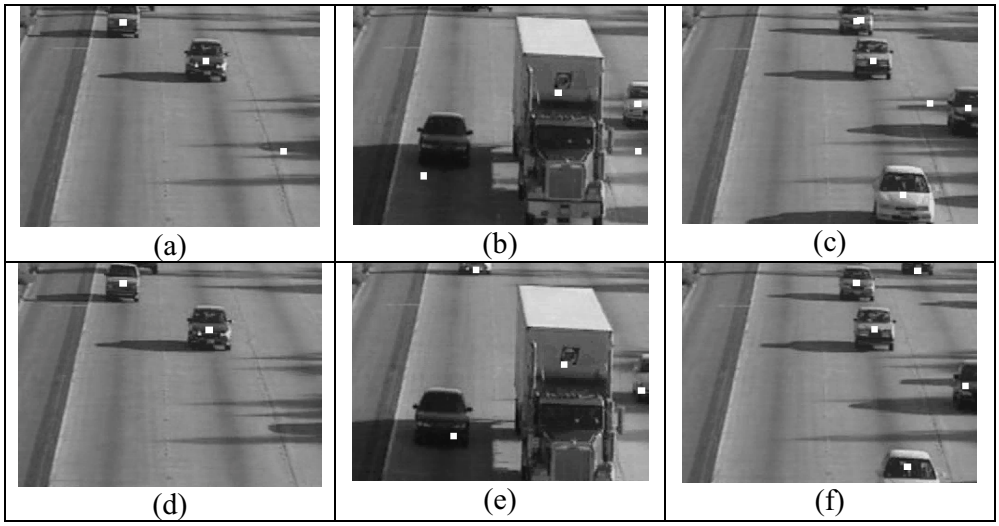


Figure 4.3.1.14. The examples of frames where our tracking system produce false alarms and the more accurately tracked vehicles in their neighboring frames of Video I (the white squares are the geometric centers of the tracked vehicles). (a) frame 114, (b) frame 364, (c) frame 414; (d) frame 115, (e) frame 365, (f) frame 415.

Table 4.3.1.2 is a set of rules of our multiple object tracking system with optimal performance for Video I and partly for Video II. For the second round of outlier strip removal, we used a loop of 7 iterations, with -2 as the width adjustment in each iteration. For shadow removal, the square size is for the small squares (e.g., square 1

in Fig. 4.3.1.12). Detailed application of these parameters and thresholds and other rules not included in this table, see Section 4.3.1.3.2.

To adaptively tune the rules to achieve the best tracking performance is our future work.

Our rule-based multiple object tracking system for traffic surveillance is comparable with some state-of-the-art tracking algorithms, in terms of high hit rate (97.92%) and low false alarm rate (1.37%). For example, in *Bai et al.* [17], the best results of the feature-based and appearance-based traffic tracking are a false alarm rate of 9.5% and a hit rate of 89.33% [17]. And *Beleznai et al*. [18] achieved a 94% hit rate but with a 37% false alarm rate for their fast mean shift model for human being tracking, which is better than blob-based approaches [19][20].

Table 4.3.1.2. An optimal set of rules of our rule-based multiple object tracking system on traffic surveillance videos I and II.

Sequenced Rules	Threshold (<=pixels)	Notes
Differencing with background	10	
Outlier removal (1st round)	16	total pixels
Hole removal (1st round)	24	total pixels
Outlier strip removal (1st round)	9*3	width*height
Hole strip removal (2nd round)	5*3	width*height
Object separation	5, 10	square size, in 2 steps
Shadow removal	7, 11	square size, in 2 steps
Outlier strip removal (2nd round)	(15:-2:3) *1	width*height, in 7 steps
Hole removal (2nd round)	36	total pixels
Hole strip removal (2nd round)	1*7	width*height

4.3.1.5 Conclusion

Multiple object tracking is known as a difficult task, especially in the presence of occlusions and background variations. Using the idea of collaborative filtering, we collaboratively extract a background from multiple independently extracted backgrounds and effectively remove spurious background pixels. With the background adaptively updated using the collaborative background extraction algorithm, multiple object tracking can be simplified and be based on the binary images from thresholding the video frames with the backgrounds. In the presence of occlusions, our rule-based tracking strategies effectively remove outliers, consolidate objects by removing holes, separate the partially occluded objects based on the

occlusion features, and remove the shadows effectively and correctly by using corner detection and removal protection rules. Empirical results on highway traffic surveillance videos show that our rule-based multiple object tracking system using the collaborative background extraction algorithm is highly practical and very accurate. It has a higher hit rate on a heavily-occluded video than that on an occlusion-free video without using our occlusion-handling strategies, and has a very low false alarm rate. Our multiple object tracking system is comparable with some state-of-the-art tracking algorithms.

References for Section 4.3.1

[1] Y-K Jung, Y-S Ho: "Multiple object tracking under occlusion conditions". In *Visual Communications and Image Processing, Proceedings of SPIE*, Vol. 4067, 2000, pp. 1011–1020.

[2] R.E. Kalman: "A new approach to linear filtering and prediction problems". *Transactions of the ASME--Journal of Basic Engineering*, Vol. 82(D), 1960, pp. 35-45.

[3] C. Chang, Ansari, R., A. Khokhar: "Multiple Object Tracking with Kernel Particle Filter". *IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[4] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, P.-J. Nordlund: "Particle filters for positioning, navigation, and tracking". *IEEE Transactions on Signal Processing*, Vol. 50(2), 2002, pp. 425-437

[5] S.-K. Weng, C.-M. Kuo, S.-K. Tu,: "Video object tracking using adaptive Kalman filter". *Journal of Visual Communication and Image Representation*, Vol. 17, 2006, pp. 1190–1208.

[6] C. Stauffer, W.E.L. Grimson: "Adaptive background mixture models for real-time tracking", *IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 2, 1999, pp. 246-252.

[7] P. Kumar, S. Ranganath, W. Huang: "Queue based fast background modelling and fast Hysteresis thresholding for better foreground segmentation". *Proceedings of the 4th International Conference on Information, Communications and Signal Processing and Pacific Rim Conference on Multimedia*, Vol. 2, 2003, pp743-747.

[8] H.-T. Chen, H.-H. Lin, and T.-L. Liu: "Multi-object tracking using dynamical graph matching", *CVPR*, Vol. 2, 2001, pp. 210-217.

[9] S.S. Intille, J.W. Davis, and A.F. Bobick: "Real-time closed-world tracking", *In IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1997, pp. 697-703.

[10] C. Rasmussen, and G. Hager: "Joint probabilistic techniques for tracking objects using multiple part objects, *In IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 1, 1998, pp. 191-196.

[11] M. Isard, and J.P. MacCormick: "Bramble: a Bayesian multiple-blob tracker", *ICCV*, Vol 2, 2001, pp. 34-41.

[12] C. Hue, J.P. Le Cadre, and P. Perez: "Tracking multiple objects with particle filtering", *IEEE Trans. on Aerospace and Electronic Systems*, vol. 38, no. 3, 2002, pp.791-812.

[13] Y. Bar-Shalom, and T. Fortmann: "Tracking and Data Association", *Academic Press*, 1988.

[14] I. Cox: "A review of statistical data association techniques for motion correspondence", *International Journal on Computer Vision*, vol. 10, no. 1, 1993, pp. 53-65.

[15] C. Rasmussen, and G.D. Hager: "Probabilistic data association methods for tracking complex visual objects", *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 23, No. 6, 2001, pp. 560-576.
[16] I.D. Reid, and D.W. Murray, "Active tracking of foveated feature clusters using affine structure", *International Journal of Computer Vision*, 18(1), 1994, pp. 41-60.
[17] L. Bai, W. Tompkinson, Y. Wang: "Computer vision techniques for traffic flow computation". *Pattern Analysis and Applications*, Vol. 7 2005, pp. 365–372.
[18] C. Beleznai, B. Frühstück, H. Bischof: "Human tracking by fast mean shift mode seeking". *Journal of Multimedia*, Vol. 1(1), 2006.
[19] A. W. Senior: "Tracking with probabilistic appearance models". *ECCV Workshop on Performance Evaluation of Tracking and Surveillance Systems*, 2002, pp. 48-55.
[20] T. Yang, Q. Pan, J. Li: "Real-time multiple objects tracking with occlusion handling in dynamic scenes". *IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 1, 2005, pp. 970-975.
[21] P. Resnick, H.R. Varian: "Recommender systems", Communications of the ACM, Vol.40(3), 1997, pp. 56-58.

4.3.2. A Progressive Edge-Based Stereo Correspondence Method

Local stereo correspondence is usually not satisfactory because neither big window nor small window based methods can accurately match densely-textured and textureless regions at the same time. In this paper, we present a progressive edge-based stereo matching algorithm, in which big window and small window based matches are progressively integrated based on the edges of a disparity map of a big window based matching. In addition, an arbitrarily-shaped window based matching is used for the regions where big windows and small windows can not find matches, and a novel optimization method, progressive outlier remover, is used to effectively remove outliers and noise. Empirical results show that our method is comparable to some state-of-the-art stereo correspondence algorithms.

4.3.2.1 Introduction

Stereo correspondence is an active research topic. The main task of stereo correspondence is to find the disparity map between a pair of images taken from two different orientations on the same scene. Accurate stereo matching remains a difficult vision problem, especially for textureless regions, disparity discontinuity, and occlusions [1]. Stereo correspondence methods roughly fall into two categories. Local stereo matching methods (window-based) capture disparities only using intensity values within a finite neighboring window. Global stereo correspondence methods such as graph cut [2] and belief propagation [1] are used to optimize the disparity map through various minimization techniques of energy that considers matching cost, disparity discontinuities, and occlusion.

For local stereo matching, small-window based matching can more accurately capture disparity in densely-textured regions, but it produces noisy disparities in textureless regions; while big-window matching produces smooth disparities in textureless

regions, but is difficult to get accurate disparities for densely-textured regions. Some algorithms have been proposed to capture disparity values for densely-textured regions, such as variable windows [3], and rod-shaped shiftable windows [4].

In an attempt to accurately match stereo for both densely-textured and textureless regions, we propose a progressive edge-based stereo matching method. The main idea is to progressively integrate big-window matching and small-window matching using the edges of a disparity map from the big-window stereo matching, so that we can match densely-textured and textureless regions at the same time. An arbitrarily-shaped windows matching, which has arbitrary shapes and orientations, is applied to the regions where a regular local stereo matching (either small window or big window matching) fails to make stereo matches.

Instead of using energy minimization based optimization, we propose an optimization method called *progressive outlier remover (POR)* to optimize the disparity map. When a disparity value is surrounded by different disparities, it will be replaced by its neighbors' average disparity when certain conditions are met. We progressively vary the distance values between the current pixel and its neighbors and use threshold values to avoid over-pruning. *POR* is similar to a diffusion-based technique [5] with respect to its smoothing out outliers.

To evaluate the performance of a stereo algorithm, a commonly-used approach is to compute the error rate with respect to some ground truth of the disparity maps [6].

$$B = \frac{1}{N}\sum_{(x,y)}(|d_C(x,y) - d_T(x,y)| > \delta_d) \cdot \quad (1)$$

where N is the total number of pixels, $d_C(x, y)$ is the computed disparity map, $d_T(x, y)$ is the ground truth map, and δ_d is a disparity error threshold.

We work on the *Middlebury* stereo data and evaluate the performance of our algorithm in terms of the accuracy for all regions, non-occluded regions, and disparity discontinuity regions of the resulting disparity maps against the ground-truth according to the *Middlebury* test bed [6].

The framework of our algorithm is in Section 4.3.2.2. The experimental design and result are in Section 4.3.2.3. Our conclusions are in Section 4.3.2.4.

4.3.2.2 Framework

As a local stereo matching method, our basic idea is to integrate big window and small window matching with the help of edges, which are extracted from the disparity map of a big window matching. An arbitrarily-shaped window matching is used for the regions where a regular local matching fails. We use a progressive outlier remover to effectively remove outliers and optimize the disparities.

4.3.2.2.1 Local Stereo Matching

A local stereo matching method seeks to estimate disparity at a pixel in one image (reference image) by comparing the intensity values of a small region (usually a square window) with a series of same-sized-and-shaped regions in the other image

(matching image) along the same scanline. The correspondence between a pixel (x, y) in reference image *R* and a pixel (x', y') in the matching image *M* is given by

$$x' = x + dis(x,y), \qquad y' = y \,. \tag{2}$$

where *dis(x, y)* is the disparity value at the point *(x, y)*.

We use root mean squared error (*RMSE*) as the matching metric

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}[R_i(x,y) - M_i(x',y)]^2} \,. \tag{3}$$

where N is the total number of pixels in a window, $R_i(x, y)$ and $M_i(x', y)$ are intensity values of pixels in the window of the reference image and matching image. The advantage of using *RMSE* is that we can use a universal threshold value for different window sizes to determine a match or non-match, without trying to choose different truncation values as other metrics such as normalized cross correlation (*NCC*) and sum of squared differences (*SSD*) do.

For each pixel in each scanline in the reference image, we seek the most similar pixel in the same scanline of the matching image, in terms of the minimum *RMSE*. If this value is smaller than a threshold value, we conclude that there is a match between the pixels and then calculate their difference along the horizontal axis as the disparity value, $dis(x,y)=x'-x$ (Equation 2). Otherwise, we report there is no match here. A disparity map has the disparity values for every pixel in the reference image.

4.3.2.2.2 Arbitrarily-Shaped Windows

We propose an arbitrarily-shaped window based stereo matching to accurately capture disparity values for densely-textured regions. Our arbitrarily-shaped-window strategy is to try out all kinds of shapes and orientations and pick the winning shape that has the lowest *RMSE*.

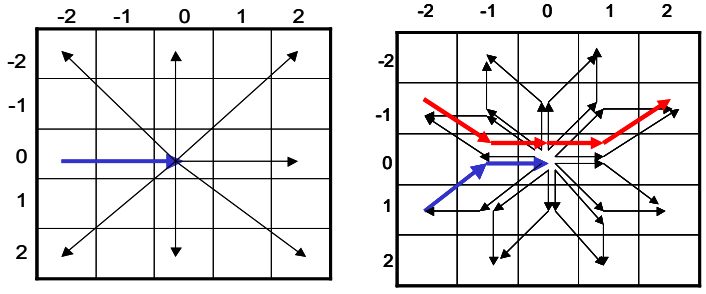


Figure 4.3.2.1. Arbitrarily-shaped windows (a) scenario A, (b) scenario B.

The arbitrary shapes or orientations come from two scenarios, scenario A and scenario B (Fig. 4.3.2.1(a)(b)). Each shape or orientation is actually a unique combination of five neighboring pixels inside a 5*5 window with the pixel (0, 0) in the middle, which is the active matching pixel.

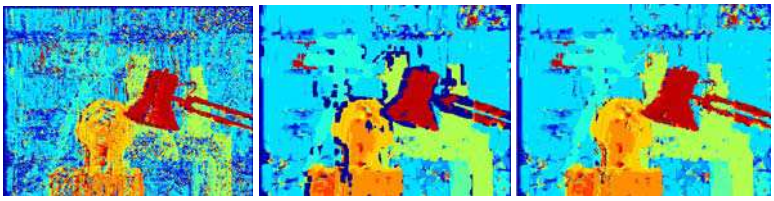
In scenario A (Figure 4.3.2.1(a)), when the first three pixels are (-2, 0), (-1, 0) and (0, 0), and our searching route for other pixels to form a unique 5-pixel combination ends

at one of the other peripheral points, we will have seven different shapes or orientations. Next, starting from another peripheral pixel and ending at a different peripheral one, we will have six (excluding the shape/orientation found in the previous search). Continuing this search until every peripheral starting pixel is tried results in a total of $\Sigma^{7}_{1}(i)=28$ shapes/orientations for scenario A. For example, the horizontal window across the point (0, 0) can be represented as (-2, 0), (-1, 0), (0, 0), (1, 0), (2, 0), where the (*x, y*) values of the points are the horizontal and vertical differences from the active matching pixel (0, 0). In scenario B (Figure 4.3.2.1(b)), we use the remaining peripheral pixels of the 5*5 square from scenario A. When our first three pixels are (-2, -1), (-1, 0) and (0, 0), we will have 15 different shapes or orientations. Taking other searching routes to form unique 5-pixel combinations, and keeping (0, 0) as the central pixel and start point and end point peripheral pixels of the square, we will get $\Sigma^{15}_{1}(i)=120$ unique shapes or orientations. For the highlighted example of Figure 4.3.2.1(b), the window is represented as (-2, -1), (-1, 0), (0, 0), (1, 0), (2, -1).

Summed from these two scenarios, we will have a total of 148 different shapes/orientations to pick a 5-pixel arbitrarily-shaped window.

We use five as the pixel number of an arbitrarily-shaped window, because three will be too small to compute reliable matching costs and seven and more will be cost prohibitive. Comparing with regular square windows, the computation time for an arbitrarily-shaped window based matching is the single 5-pixel matching time multiplied by 148 (5*148), which is equivalent to matching with a square window of size 27 (27*27). By applying the arbitrarily-shaped windows only for the regions where a regular window based matching can not find matches (less than 10%), the complexity is greatly reduced.

Figure 4.3.2.2 illustrates the effect of using arbitrarily-shaped windows on the stereo data *Tsukuba*, and its combinational usage with a regular square window stereo.



(a) (b) (c)

Figure 4.3.2.2. (a) Arbitrarily-shaped window (*Wa*) matching for the stereo data Tsukuba, (b) window 7*7 (*W7*) matching, (c) window 7*7 + arbitrarily-shaped window (*W7+Wa*) matching.

4.3.2.2.3 Progressive Edge-Based Stereo Matching

The main steps of our progressive edge-based stereo matching are illustrated in Figure 4.3.2.3. As an example of our stereo matching on the stereo data *Teddy*, Figure 4.3.2.3(a) is the disparity map from a small-window matching (*win_small*) of size 3*3 (*W3*); (b) is the disparity map from a big window matching (*win_big*) of size 25*25 (*W25*); (c) is the disparity map from the arbitrarily-shaped windows matching

(*win_arbi, or Wa*); (d) is *W3+Wa* (*W3*, plus *Wa* where *W3* can not make matches); (e) is *W25+Wa*; (f) is the edges of the *W25+Wa* optimized by the *POR*; (g) is the disparity strips combined from (d) and (e) around the edges in (f); (h) is the smoothed disparity map from (g), (i) is the final disparity map optimized from the *POR* optimization method.

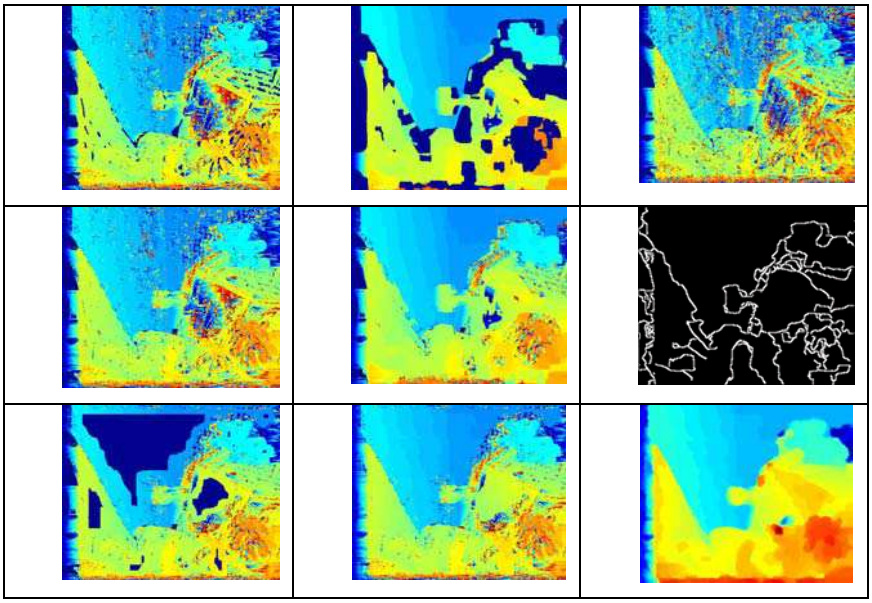


Figure 4.3.2.3. An illustration of our edge-based stereo correspondence on the stereo data Teddy (top row) (a) W3, (b) W25, (c) Wa; (middle row) (d) W3+Wa, (e) W25+Wa, (f) edges of W25+Wa; (bottom row) (g) strips of W3+Wa and W25+Wa around the edges, (h) smoothed disparities between strips, (i) final disparity map after POR optimization.

We use the optimal edge detector *Canny* to detect edges [7]. We use the command *edge(image, 'canny', k)* in MATLAB to get the edge file for the input disparity map image. A smaller *k* value will generate more edges in the binary output edge file, in which 1 represents an edge and 0 otherwise.

When combining the big window and small window matching, with arbitrarily-shaped windows used for regions where a big window or small window can not make matches, we use the disparity values from *small window matching* for the strips around the edges; and use the disparities from *big window matching* for the strips away from the edges (next to the *small window matching* strips). The width of *small window matching* strips at each side of the edges and that of the neighboring *big window matching* strips are W_{strip}=*[size(win_big)-size(win_small)]/2+1*.

We enforce the disparity continuity between the strips of *big window matching* using a disparity averaging scheme. Suppose a pixel *(x, y)* inside the region is to be smoothed, the disparity value *dis(x, y)* depends on the closest disparity values of four directions on its neighboring strips. Given horizontally left and right disparities

dis(x1,y) and *dis(x2, y)*, and vertically above and below disparities *dis(x, y1)* and *dis(x, y2)*), we calculate the disparity value *dis(x,y)* by $dis(x,y)=\frac{1}{2}[dis_x(x,y)+dis_y(x,y)]$, where

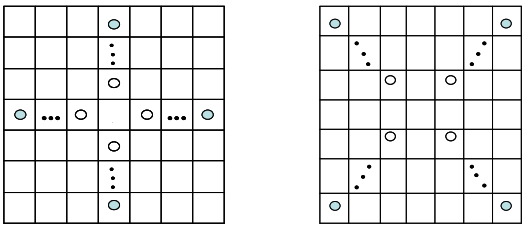
$$dis_x(x,y)=\frac{dis(x_2,y)-dis(x_1,y)}{x_2-x_1}(x-x_1)+dis(x_1,y),$$

$$dis_y(x,y)=\frac{dis(x,y_2)-dis(x,y_1)}{y_2-y_1}(y-y_1)+dis(x,y_1)\cdot$$

4.3.2.2.4 The Progressive Outlier Remover Optimization

Our *Progressive Outlier Remover (POR)* optimization is based on the disparity continuity assumption: in a small region, when a disparity value is greatly different from its surroundings, it is deemed as an outlier and should be replaced or optimized.

Illustrated in Figure 4.3.2.4, for each value in the disparity map, we compare it with four equally-distanced neighbors in four directions, separately, one kind of neighbors are directly above, below, left, and right neighbors (Figure 4.3.2.4(a)), and another kind are four corners of a square where the current pixel is centered (Figure 4.3.2.4(b)).



(a) (b)

Figure 4.3.2.4. An illustration of POR optimization: a disparity outlier is replaced with an average of its four neighbors' disparities in either scenario A (a) or scenario B (b), and the neighborhood distance is adjustable.

When the disparity of the central pixel is not equal to any of its neighbors' disparities, and its difference from the average of the neighbors' disparities is bigger than a threshold, it will be replaced by the average of the neighbors' disparities. The threshold *T* is proportional to the product of the neighborhood distance *d* to the central pixel and the standard deviation σ of the four neighbors' disparity values

$$T=k\times d\times\sigma,\quad \sigma=\sqrt{\frac{1}{N}\sum_{i=1}^{N}x_i^2-(\frac{1}{N}\sum_{i=1}^{N}x_i)^2}\cdot \tag{4}$$

where N = 4, and k takes a value of 1 or 0.5. A k value of 1 represents relatively stricter thresholds than that of 0.5.

The *POR* optimization algorithm takes two parameters, maximum neighborhood distance *d* (a value usually from 2 to 20), and a decremental rate *R* (a value of 2/3, 1/2, or 2/5) for getting decrement iteration numbers in different rounds of iterations. For each round *i* (i=1, 2, 3, ……), we calculate the iteration number n_i, with initialization of $n_1=d$, and the iterations will not stop until $n_i=1$.

$$n_i=\lfloor n_i\times R^{i-1}-0.5\rfloor+1\ . \tag{5}$$

For example, when we have *(d, R)*=(20, 2/3), we will have 8 rounds of optimizations, but with *R=1/2* and *R=2/5*, we will have 6 and 4 respectively. For each round of iterations, we have 1 and 0.5 as the *k* values alternatively, i.e., we have (20, 1), (20, 0.5), (8, 1), (8, 0.5), (3, 1), (3, 0.5), and (1, 1) as the (n_i, *k*) combinations for *(d, R)*=(20, 2/5), and we do not have (1, 0.5) for n_i =1. For each round, with the iteration number n_i, the *POR* algorithm will have iteration *i* from 1 to n_i, each of which has the neighborhood distance of *i*, and has the threshold for outlier removal of $k*i*\sigma_i$ defined in Equation 4.

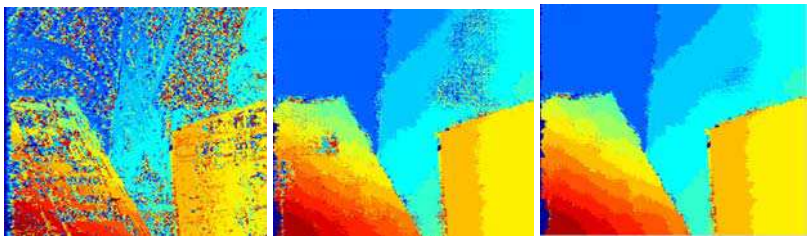
The complete *POR* algorithm is in Figure 4.3.2.5. An example of applying this algorithm on the stereo image *Venus* is shown in Figure 4.3.2.6.

***Algorithm*: progressive outlier remover (*POR*) (d, R, dis(x,y))**

for each n_i of (i=1, n_1=d; $n_i \geq 1$; $n_i = \lfloor n_i * R^{i-1} - 0.5 \rfloor + 1$, i++)

```
{  for k_round=1:2
  { if (k_round==1) k=1;
    else k=0.5;
    for (n=1; n<= n_i; n++)
    {  d=n (neighborhood distance)
       T_A=k*d*σ_A, T_B=k*d*σ_B (thresholds for A, B scenarios)
       for each pixel dis(x, y) in the disparity map,
        {  if(dis(x,y)≠ ∀∈{dis(x-d,y),dis(x+d,y),dis(x,y-d),
             dis(x,y+d)} && abs(dis(x,y)-μ_A)>T_A)
          { dis(x,y)=(dis(x-d,y)+dis(x+d,y)+dis(x,y-d)
            +dis(x,y+d))/4
          }
          if(dis(x,y)≠∀∈{dis(x-d,y-d),dis(x+d,y+d),dis(x+d,y-d),
            dis(x-d,y+d)} && abs(dis(x,y)-μ_B)>T_B)
          { dis(x,y)=(dis(x-d, y-d)+dis(x+d, y+d)+
            dis(x+d, y-d)+dis(x-d, y+d))/4
          }
       }
    }
}
```

Figure 4.3.2.5. The Progressive Outlier Remover (POR) optimization algorithm.



(a) **(b)** **(c)**

Figure 4.3.2.6. Applying the POR optimization on the stereo data Venus (a) disparity map of W3+Wa (win_size 3*3 +arbitrarily-shaped windows), (b) (c) after the first two rounds of POR optimizations with d=14, R=1/2.

4.3.2.3 Experimental Design and Results

We work on four *Middlebury* stereo images, *Tsukuba*, *Venus*, *Teddy*, and *Cones* for quantitative evaluation, which are the benchmark data for stereo correspondence algorithms [8]. We evaluate our algorithm in terms of the percentage of bad pixels, i.e., pixels whose absolute disparity error is greater than a threshold (such as 1 and 0.5). We calculate percentages for (1) pixels in non-occluded regions, (2) all pixels, and (3) pixels near disparity discontinuities, and ignore a border of 10 pixels for *Venus*, and 18 for *Tsukuba* when computing statistics, according to the evaluation standard on the *Middlebury* stereo data [6].

For window based stereo matching, we use *RMSE* as the cost metric, and use 15 as a universal cut off value for determining a correspondence, based on our preliminarily experiments. As described in Section 4.3.2.2, our big window and small window matching are actually *win_big+win_arbi* and *win_small+win_arbi*, where the arbitrarily-shaped window based matching is applied to the regions that a regular big window or small window matching fail to make matches. When optimizing the disparity map using the *POR* optimization algorithm, we use different parameters of (*d*, *R*) for different stereo data, according to the distribution of densely-textured and texture-less regions.

The stereo data *Teddy* has a big textureless area, which a regular local stereo algorithm has difficulty dealing with. By using a big window match of size 25, optimized by *POR* of *d=12*, the big hole in the textureless regions of the disparity map is well smoothed (Figure 4.3.2.3). For the representative densely-textured data *Tsukuba*, we use relatively small window sizes (with big window size of 5) and small parameters for the *POR* optimization (d=3), to avoid the loss of the accurate disparities for the delicate textures.

The overall evaluation of our algorithm is in Table 4.3.2.1, Table 4.3.2.2, and Figure 4.3.2.7. In Table 4.3.2.1, we find that there is apparent improvement of stereo correspondence using the edge-based strategy over that without using it, the later of which simply uses a *win_small+win_arbi* matching and gets it optimized by the *POR* algorithm.

Table 4.3.2.1. Improvement of using progressive edge-based stereo matching over without using edge-based strategy (in terms of percentage of bad pixels for non-occluded, all and disparity discontinuity regions, with threshold of 1).

	Tsukuba			Venus		
	nonocc	All	disc	nonocc	all	disc
w/o edge-based	2.98	4.92	15.1	2.47	3.48	27.5
edge-based	**2.73**	**4.65**	**13.9**	**2.25**	**3.24**	**27.4**
	Teddy			Cones		
	nonocc	All	disc	nonocc	all	disc
w/o edge-based	14.5	22.4	33.0	9.78	17.5	21.3
edge-based	**14.3**	**22.1**	**30.2**	**7.63**	**16.1**	**19.7**

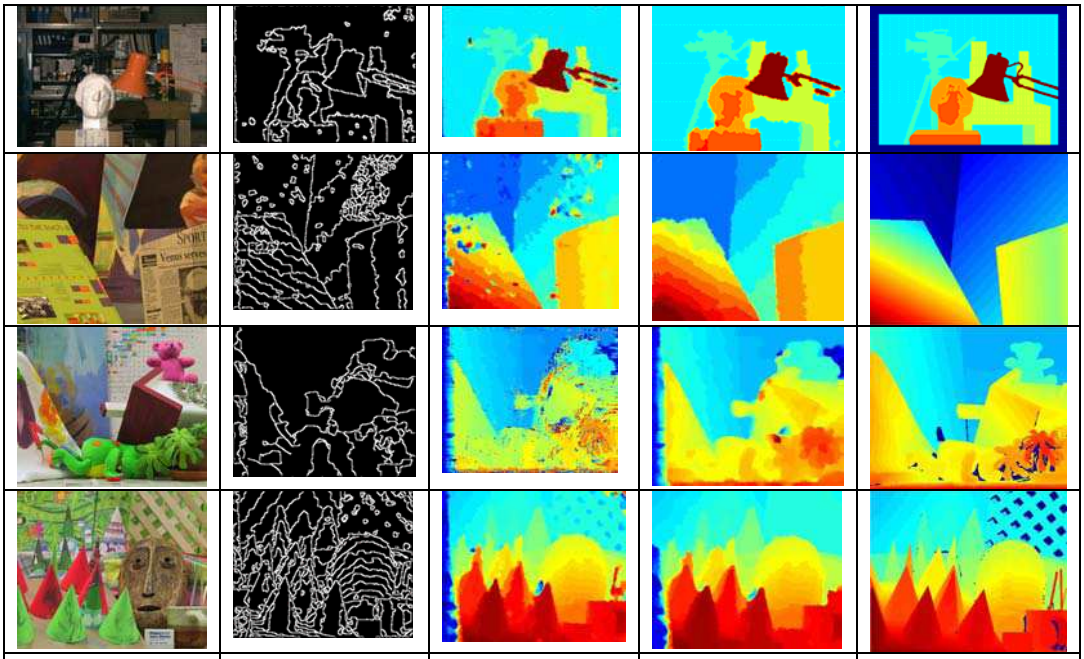
By the time of submission, the average rankings of our algorithm on the *Middlebury* stereo evaluation system [8] are No. 16 for error threshold of 0.5, and No. 22 for error threshold of 1, out of 29 submissions to the system, most of which are results from existing state-of-the-art algorithms. Compared with other disparity optimization methods, our algorithm is better than scanline optimization [6] and comparable to graph cuts using alpha-beta swaps [9] and dynamic programming [10] on the new version of *Middlebury* evaluation. On the previous version of *Middlebury* evaluation data, our algorithm is better than other window-based stereo correspondence algorithms such as the pixel-to-pixel algorithm [11], the discontinuity preserving algorithm [12], and the variable window algorithm [3].

With the threshold of 0.5, our progressive edge-based stereo matching has the average rankings of No. 14 for the non-occluded regions and all regions, but has the average ranking of No. 22 for the disparity discontinuity regions. We plan to improve this algorithm in our future work, especially its performance in matching the disparity discontinuity regions.

The parameter settings of our progressive edge-based stereo matching can be unified for different stereo data by analyzing the distributions of highly textured and textureless regions. We will investigate this in the future. It will also be interesting to combine our local stereo method with global optimization algorithms such as graph cut and belief propagation.

Table 4.3.2.2. Overall evaluation of our algorithm on the Middlebury data (in terms of percentage of bad pixels for non-occluded, all, and disparity discontinuity regions; the subscripts of the results are our rankings amongst other state-of-the-art algorithms on the Middlebury stereo system, with thresholds 1 and 0.5).

	Tsukuba			Venus		
	nonocc	All	disc	nonocc	all	disc
Thre=1	2.73_{19}	4.65_{20}	13.9_{24}	2.25_{21}	3.24_{21}	27.4_{27}
Thre=0.5	8.26_{5}	10.4_{6}	23.0_{21}	8.57_{13}	9.67_{13}	33.9_{25}
	Teddy			Cones		
	Nonocc	all	disc	nonocc	all	disc
Thre=1	14.3_{22}	22.1_{22}	30.2_{26}	7.63_{20}	16.1_{20}	19.7_{22}
Thre=0.5	24.3_{20}	32.2_{22}	43.0_{24}	15.0_{17}	23.0_{17}	28.0_{20}



Reference image	Edges of win_big + win_arbitrary	win_big and win_small combined	Disparity map after the POR optimization	Ground truth disparity maps

Figure 4.3.2.7. The results of our progressive edge-based stereo matching algorithm (from top to down: Tsukuba, Venus, Teddy, and Cones. Same color on different maps does not necessarily represent the same disparity).

4.3.2.4 Conclusions

Stereo correspondence is a difficult vision problem, especially for textureless regions, disparity discontinuity, and occlusions. In order to address the problem that regular window based stereo matching methods fail to make accurate matches for densely textured and textureless regions at the same time, we propose a progressive edge-based stereo matching method to unify big window matching and small window matching with the help of edges. The edges are extracted from the disparity map of the big window matching using an optimal edge detector. An arbitrarily-shaped window based matching is used for the regions where a regular big window or small window matching can not find matches. Instead of using an energy-minimization based optimization, we propose a novel optimization algorithm called *progressive outlier remover (POR)*, which progressively replaces an outlier disparity value with the average of its surrounding ones when certain constraints are met, and effectively removes outliers and enforces disparity continuity. Experiments on the standard *Middlebury* stereo data show, that our progressive edge-based stereo matching method performs comparable with some state-of-the-art stereo matching algorithms in terms of matching accuracy for non-occluded regions and all regions.

References for Section 4.3.2

1. Sun, J., Zheng, N-N., and Shum, H-Y.: Stereo Matching Using Belief Propagation, *PAMI*, Vol. 25(7) (2003)
2. Kolmogorov, V., Zabih, R.: Computing Visual Correspondence with Occlusions using Graph Cuts, *ICCV*, (2001)
3. Veksler, O.: Fast Variable Window for stereo correspondence Using Integral Images, *CVPR*, (2003)
4. Kim, J.C., Lee, K.M., Choi, B.T., Lee, S.U.: A Dense Stereo Matching Using Two-Pass Dynamic Programming with Generalized Ground Control Points, *CVPR*, (2005)
5. Scharstein, D., Szeliski, R.: Stero Matching with Non-linear Diffusion, *IJCV*, Vol. 28(2) (1998) 155-174
6. Scharstein, D., Szeliski, R.: A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms, *IJCV*, Vol. 47(1) (2002) 7-42
7. Canny, J.: A Computational Approach To Edge Detection, *PAMI*, Vol. 8 (1986) 679-714
8. Middlebury stereo, http://www.middlebury.edu/stereo
9. Boykov, Y., Veksler, O., Zabih R.: Fast Approximate Energy Minimization via Graph Cuts, *PAMI*, Vol. 23(11) (2001)
10. Bobick, A.F., Intille, S.S.: Large Occlusion Stereo, *IJCV*, Vol. 33(3) (1999)181-200
11. Birchfield, S., Tomasi, C.: Depth Discontinuities by Pixel-to-Pixel Stereo, *ICCV*, (1998)
12. Agrawal, M., Davis, L.: Window-Based Discontinuity Preserving Stereo, *CVPR*, (2004)

4.4 3D Video Compression, Delivery and Playback

This part of the report describes novel algorithms and technologies related to effective compression, delivery and playback of multi-view digital video sequences.

4.4.1 Multi-View Video

CCCD cameras are often used for security and surveillance purposes and more than often we have multiple cameras setup to view the different parts of the areas supposed to be secured and under surveillance. In the case of coastline, we may have a setup of arrays of hundreds of cameras and to effectively control and use these cameras is one of the biggest tasks. We may have a best infrastructure but to get maximized and effective benefits from that infrastructure is a challenging task.

4.4.1.1 CCCD Camera Overview

The navigation in the multi-camera setup needs an easy to use mechanism to control cameras as well as getting frame data from these cameras in the form of a stream. One of the major tasks is to develop a customized APIs for these cameras so that we are able to completely control the cameras as well as able to get our required functionality. Camera vendors often provide SDKs for their cameras so that the developers can build customized applications to use the cameras. The problem with the vendor SDKs provided by the vendors of the CCCD cameras is that they provide high level functionality which is often insufficient to do the task at hand. The goal of

our project is to allow navigating the cameras and controlling of multiple camera views such that the user has the freedom of choosing the view in the desired format, resolution and frame rate for a particular camera or a number of cameras.

The Camera interface APIs are designed to effectively control the cameras' stream and format. It can activate a particular camera or multiple cameras and it can also save the stream from these cameras to the storage media.

4.4.1.2 Video Format

These CCCD cameras support the data formats RGB and Y800 but in order to use the high end encoders that support the latest video encoding standards like H.264, we have to convert the image buffer stream to YUV420 format. YUV format can be used by these encoders effectively. The only limitation is the bandwidth of the media over which these streams will be transmitted. Since the raw format (YUV) takes a lot of storage space, therefore, in some cases we may have to restrict the frame control in order to avoid the frame skipping.

4.4.1.3 Camera Initialization

We can initialize as many cameras as we want by using the cameras' unique names which is built in every camera. The initialization of the cameras has nothing to do with the actual functioning i.e. frame grabbing of the cameras. It only prepares the cameras' buffers and initializes the filters of the cameras.

4.4.1.4 Start/Stop Camera

The cameras can be started or stopped by specifying the camera number. Camera number is the index of the camera that specifies the order in which cameras are initialized. The client application has to keep track of the camera index along with the camera name or it can also be specified by using a configuration file. We can also start/stop all cameras at once without using their names.

4.4.1.5 Frame Rate

The default frame rate is 30 frames per second. We can change the frame rate of a specific camera or all of the cameras through this API interface. The new frame rate will be applied to a camera or all cameras provided it is supported by the camera hardware. Otherwise the default frame rate will be applied.

4.4.1.6 Resolution

The default resolution is 640x480. We can change the resolution of a specific camera or all of the cameras through this API interface. The new resolution will be applied to a camera or all cameras provided it is supported by the camera hardware. Otherwise the default resolution will be applied.

4.4.1.7 Frame Grabbing

This is the most important task achieved by the camera API. Grabbing frame data is supported by converting the input stream to YUV format and this YUV frame data is returned frame by frame along with its width and height. Frame data can only be obtained by specifying a specific camera index. At the initialization time, the format of the camera stream can be specified as one of RGB or Y800. This API then converts the stream (frame data one by one) and returns it to the client along with the width and height of the frame, thus reducing some calculation workload on the part of client.

4.4.1.8 Saving Frame Data

The frame data can be saved to a file on the storage media by specifying the camera index and file name along with the path. Frames are saved in YUV format but can be encoded to save the space on storage media. Saved data can be used later for various purposes. Frame data can only be saved for individual cameras i.e. frame from different cameras cannot be merged into one stream or file on storage media. It can be done at a later stage to show different views from different cameras embedded inside a single window.

4.4.1.9 Implementation

This project was implemented by using the IC Imaging Control (version 3.0.4) camera SDK and we built a library of functions to do the different tasks explained earlier. Our first task was to get the data in YUV420 format which we did successfully since the SDK does not provide the required format, we had to manually transform the format of the camera stream. This transformation occurs inside the implementation of a filter which is called transform filter. We used IPP APIs to implement this transformation. IPP provides a very efficient implementation of transformation (conversion) functions from one format to another format. By using IPP APIs, we made sure that there is no significant performance hit while converting the image streams from one format to another format.

4.4.1.10 Camera Synchronization

The projected scenario of multiple cameras possibly consisting of hundreds of cameras may introduce the lag in the visualization of streams from those cameras which are started later. We solved this problem by introducing relative delay in the starting time of the cameras. Since these cameras don't have the inbuilt mechanism of synchronization with each other, therefore we were able to reduce this lag up to a few hundreds of milliseconds which is minimum possible lag that may occur in the image streams for multiple cameras.

4.4.1.11 Test

The test application for the camera APIs makes calls to the library functions while at the front end we integrated the input functionality with Wii controller. We have four CCCD cameras mounted on a camera stand in the form of an array. The limitation is due to the firewire connections in the desktop computer since there are only four ports in a firewire card. A further limitation for multiple firewire cards on a single computer could be the bus bandwidth of the mainborad.

The Wii controller was used to select the one camera input out of four cameras by just moving the Wii controller in up, down, right or left. As it is a prototype, we designated four cameras labeled as up, down, right and left. So, by just moving the Wii controller, a particular camera stream can be selected which is displayed live on the screen. Further by making GUI for this application, proper controls can be built to change the settings of the cameras and saving camera streams to storage media.

4.4.1.12 Program Structure

4.4.1.12.1 Data Structure

CameraElem is the basic data structure that contains the properties the camera, we want to control.

```
// Camera specific data for individual cameras
typedef struct _CAMERA_DATA
{
        unsigned int camera_number;
        const char* camera_name;
        const char* resolution;
        unsigned int fps;
        DShowLib::Grabber m_Grabber;
        DShowLib::FrameHandlerSink::tFHSPtr m_pSink;
        CConversionFilter m_Filter;
} CameraElem;
```

4.4.1.12.2 Library Functions

Below are the functions that our library provides.

```
/** This function must be used to initialize the cameras. **/
bool InitCameras(const char** ppNames, unsigned int count);

/** This function must be called after the cameras are initialized.  This function starts
the camera which starts grabbing frames.
**/
bool StartCamera(unsigned int cameraNumber, bool all=false);

/** This function MUST be called befor StratCamera(). This function sets the frame
rate of the specified camera.
**/
bool SetFrameRate(unsigned int cameraNumber, double fps, bool all=false);

/** This function MUST be called befor StratCamera(). This function sets the
resolution of the specified camera.
**/
```

```
bool SetResolution(unsigned int cameraNumber, const char* resolution, bool
all=false);

/** This function returns the frame data of one frame for a specified camera along
with the width and height of the frame
**/

bool GetFrame(unsigned int cameraNumber, unsigned char** ppFrame, int* pWidth,
int* pHeight);

/** This function starts writing frame data to the file.
**/
bool StartWriteToFile(unsigned int cameraNumber, const char* fileName);

/** This function stops writing frame data to the file.
**/
bool StopWriteToFile(unsigned int cameraNumber, bool all=false);

/** This function stops the cameras.
**/
bool StopCamera(unsigned int cameraNumber, bool all=false);

/** THIS FUNCTION IS FOR TEST PURPOSE.
 * It starts to display video.
**/
bool ShowVideo(unsigned int cameraNumber);

/** THIS FUNCTION IS FOR TEST PURPOSE.
 * It stops displaying video.
**/
bool StopVideo(unsigned int cameraNumber);
```

4.4.1.13 Schematic Diagram

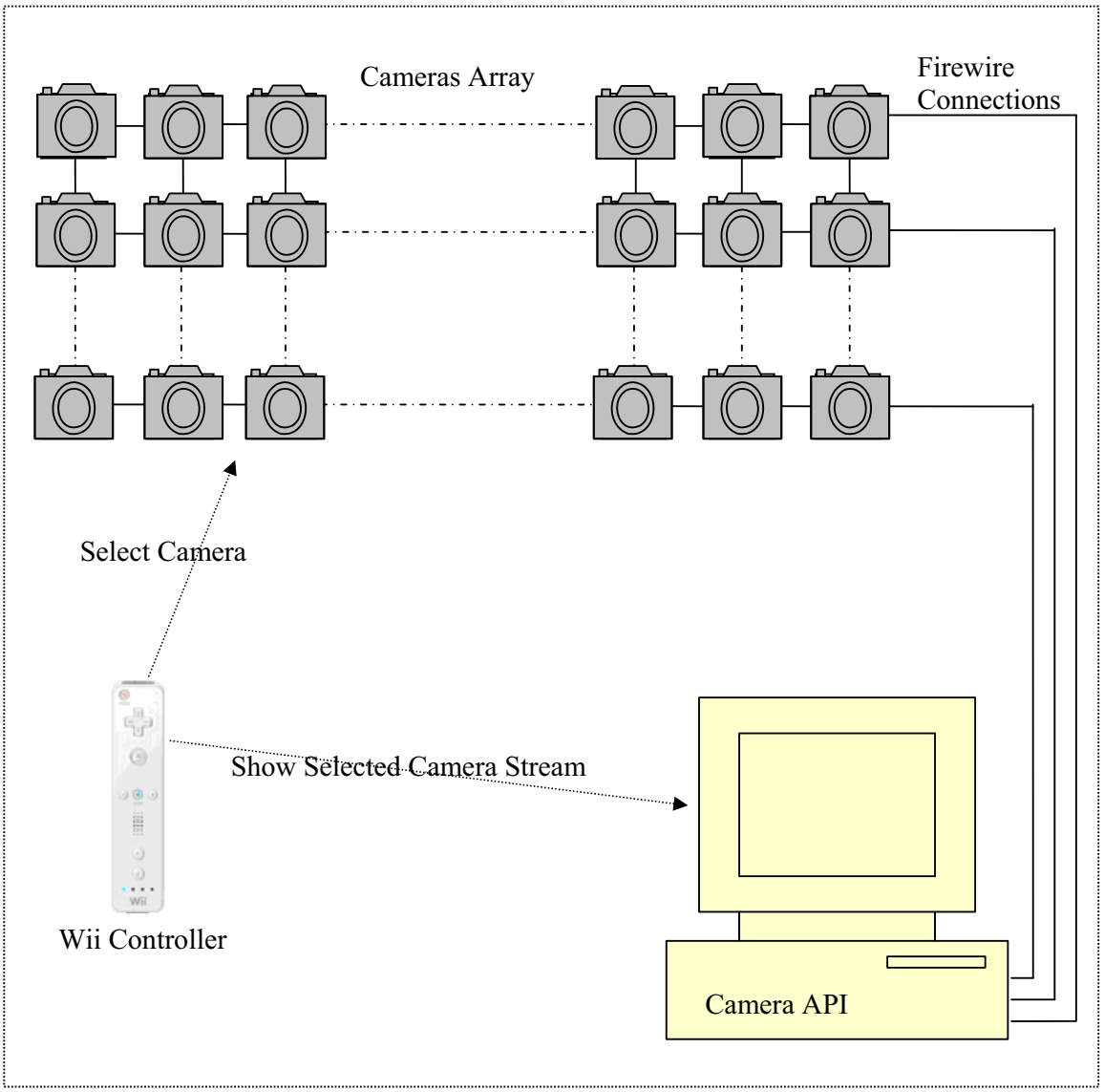
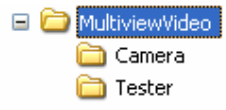


Figure 4.4.1. The schematic diagram of the environment setting.

4.4.1.14 Directory Hierarchy

MultiviewVideo.zip contains the source code of the library APIs as well as source code of the test application.



The "Camera" folder contains the following source code files for the library API.

1. stdafx.h
2. Camera.h
3. Camera.cpp

4. ConversionFilter.h
5. ConversionFilter.cpp

When compiled in Microsoft Visual Studio 2005, it will generate "Cemear.lib".

The "Tester" folder contains the following source code file for the test application.

1. Tester.cpp

This tester application uses the "Camera.lib" generated in earlier step and it has to be included in the path of the client or test application.

Dependencies:
1. Intel IPP (Integrated Performance Primitives)
2. IC Imaging Control

4.4.2 Smart Video Encoding

H.264 is relatively more complex encoder than prior standards such as MPEG-2 and H.263 which makes H.264 video encoding on mobile and wireless devices expensive. In this paper we present an approach to perceptual quality enhancement based on content awareness relative to points of salient locations in video pictures. We detect the visually salient regions of interest and modify the rate control through quantization based not only on buffer fullness but also on the level of detail in the picture. The results show that by making small quality compromises in regions of the picture that are perceptually less important and in such a way that is intended to be minimally perceptible, we can improve the perceived quality of the selected regions in the video without affecting the bitrate. The experimental results show that the proposed method improves perceptual quality of salient regions and decreases the bitrate with some loss in overall PSNR. Furthermore, the proposed approach is very simple and therefore, it can be used efficiently for video coding on low complexity devices such as mobile phones.

4.4.2.1 Introduction

H.264/AVC has been designed to provide a technical solution appropriate for broadcast, storage device, and conversation service over wireless networks, VOD or multimedia streaming services. Recent developments in video encoding standards such as the H.264 have resulted in highly efficient compression [1]. Experimental results show that H.264's coding performances overcome MPEG-4 at low bit rates [2]. The current video compression algorithms also support multiple reference frames for motion compensation. As the number of reference frames is increased, the complexity increases proportionally.

Resource constrained devices typically manage the complexity by using a subset of possible coding modes thereby sacrificing video quality. This quality and complexity relationship is evident in most video codecs used today. Most H.264 encoder implementations on mobile devices today do not implement the standard profiles fully

due to high complexity. For example, Intra 4x4 prediction is usually not implemented due to complexity. In [6] complexity is reduced by using machine learning algorithm to predict Intra MB coding. In addition several rate control techniques have been proposed to achieve balance among efficiency and complexity e.g. mode selection and post-quantizer control. Proper rate control can significantly improve the performance by reducing time-out effects, packet loss thus enhance the video quality and guarantee quality of service (QOS) [7][8]. The rate control algorithms that appear in literature primarily focus on achieving target bitrate and do not consider the content and human perception. Similar work has been done in model based coding where content modeling drives bit allocation [11]. Content modeling such as face detection is computationally expensive and cannot be used in mobile devices.

There exist technologies for video streaming in networks with fluctuating bandwidth that define the regions of interest in a video. MPEG-4 selective enhancement [10] is used in the enhancement layer of MPEG-4 FGS (Fine Grained Scalability) in order to stream better quality of video within selected image regions. However, MPEG-4 selective enhancement does not provide quality improvement for the base layer. FGS-MR video encoding [9] uses MR (Multi-Resolution) frames based on MR masks to improve the rate distortion performance. However, this process essentially introduces the complexity in the encoding process since canny edge detection is a resource consuming process.

In this paper we take into account the perceptual quality of the image and the areas in that image that receive most user attention. We try to improve the quality of that area and compromise the perceptual quality of the remaining image. It provides us the opportunity to concentrate more on the area in the image where more likely user attention will be focused. By paying more attention to this area we enhance the perceptual quality of that area while neglecting and sometimes sacrificing the perceptual quality of the areas where most likely user attention will not be focused.

The proposed perceptual enhancement technique in this paper uses detail detection and based on the amount of detail, the candidate area for user attention is identified and quantized according to the information received beforehand. Thus, given a low bandwidth (e.g. in bitrate), the proposed technique can improve the perceptual quality of the desired areas in a video which usually is less than the rest of the video's part and it is quantized with some perceptual loss in the visual quality of the video. Hence, our technique can improve the bitrate with very small loss in PSNR (less than 0.5 dB). Consequently, this technique can achieve lower bitrate video encoding compared to other methods discussed above. Moreover, this technique processes the video in a manner that is transparent to the decoder. No additional parameters are required for the decoder to decode each frame.

4.4.2.2 Proposed Approach

Figure 4.4.2.1 shows the schematic diagram of rate control in JM reference software. The blue block shows the additional information we are using in the rate control algorithm in order to implement our technique. H.264 standard uses an efficient rate control algorithm with HRD (Hypothetical Reference Decoder) considerations. It is operated at both frame and basic unit level. The QP of the basic unit is decided using the decided initial frame QP. Encoder's configuration file specifies both the size of

basic unit as well as initial frame QP. If the size of basic unit is 1, the rate control operates on each MB.

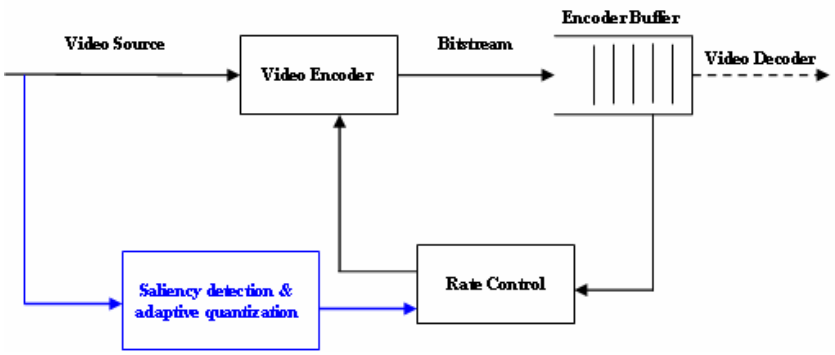


Figure 4.4.2.1. Rate control in H.264 with proposed modification illustrated by blue part.

It must be noted that our approach is actually an improvement of the rate control algorithm in many aspects, as will be shown later in the paper. The modification proposed by our approach actually allows us to have beforehand information about the MBs that are going to be encoded. However, it is first necessary to understand rate control in H.264 in order to grasp fully the functioning of our approach.

The rate control mechanism in JM [4] consists of three components:

(1) *GOP level rate control*: GOP level rate control calculates the available bits for the remaining frames in the GOP and initialize the QP of instantaneous decoding refresh (IDR) frame.

(2) *Picture level rate control*: In this level quantization step Q_{step} is computed by the quadratic model and then used to perform R-D Optimization (RDO) for each MB in the frame.

(3) *Basic unit level rate control*: the basic unit is defined as a picture slice, MB row or a set of MBs. A linear model is used to predict MAD of the current basic unit in the frame and a quadratic R-D model is used to calculate the QP which is used for RDO in the basic unit. The basic unit level rate control is to obtain a good tradeoff between the picture quality and bit fluctuation.

4.4.2.3 Implementation

Since the target appliances for our technique are wireless devices, we decided to use only one reference frame with frame rate of 30 fps. To further reduce the complexity we disabled all inter block search modes except that of 16x16 as well as RD Optimization [5]. The size of basic unit [3] is 11. All of our simulations were performed on 300 frames of QCIF video sequences. We also tried this approach on CIF video sequences but the results were not encouraging. We used the H.264 reference software implementation JM10.2.

Our technique is based on the fundamental observation that in mobile devices usually a very small portion of the video requires user attention. The semantically/visually

important regions of the video frame are detected by the edge detection method. Thus, our approach is a two step process:
Define the regions of semantic/visual interest within each basic unit of the frame.
Use the information gained in the above step to properly define the quantization step in the rate control algorithm.

In the next section, we first describe the edge detection to define the region of interest. Next we describe an effective method for adjusting the quantization step for efficiently encoding the frame that results in reducing the complexity of the encoder.

4.4.2.3.1 Low Complexity Method for Detecting Perceptually Important Regions

Edges characterize boundaries and are therefore a problem of fundamental importance in image processing. Edges in images are areas with strong intensity contrasts – a jump in intensity from one pixel to the next. By detecting the difference in intensity contrast in an MB, we can predict activity in that MB. In other words, this activity may well be an indication of having areas that attract viewers' attention. We identify the intensity of activity in an MB with reference to a threshold and mark as important. The thresholds are determined empirically by evaluating a set of video sequences. It may be noted that the technique we used is different from the traditional edge detection techniques that use Sobel or Laplacian methods and are computationally extensive. Our technique is simple and computationally effective. It is based on the fundamental observation that the areas of the image that form important features for the semantic/visual understanding of the image have strong intensity contrast. The algorithm for detecting areas of importance works as follows:

```
Algorithm: find_perceptually_relavent_block
Input: Block, pixelDifference, pixelThreshold
Output: PRBlock
edge ← 0; count ← 0; PRBlock ← 0
∀ 0 ≤ row ≤ 14, 0 ≤ col ≤ 14
    if (Block[row][col]- Block [row][col+1]) > pixelDifference  then
        count ← count + 1
    if (Block [row][col]- Block [row+1][col]) > pixelDifference  then
        count ← count + 1
if (count > pixelThreshold)  PRBlock = 1
return PRBlock
```

The pixelDifference and pixelThreshold values are determined empirically and are tuned to detect blocks of interest. We have used an optimized implementation for Intel processors to detect this intensity which is a highly efficient process and has almost negligible effect on the performance of the encoder. An MB is marked as a candidate for applying better quantization with reference to a particular threshold that is specified beforehand and it depends upon the pixel value. Once these MBs are

identified, we keep track of the total number of such MBs until a basic unit is processed.

The identification of such MBs having areas of content activity is a key to our approach which is based on the preconfigured threshold. The edge detection also plays a very important role in the content based analysis of the video scenes and is used in the applications employing similar techniques for RD optimization or bitrate reduction. A Canney edge detector based approach to identifying perceptually important areas was presented in [9]. This approach is computationally expensive and not useable on low complexity devices such as mobile phones.

4.4.2.3.2 Adaptive Quantization

We adjust the QP by increasing or decreasing it and this process is repeated for every basic unit. We specified a moderate threshold for varying quantization. If out of 11 MBs in a basic unit, 6 edges are detected, we reduced the QP by 3, otherwise increase the QP by 3. The QP is increased or decreased after it is determined for the current basic unit. Therefore, the information obtained about this basic unit in the earlier step is really helpful while the encoder is performing this step.

This is final step that decides the bitrate and perceptual quality of the video. On the one hand, when an edge is detected, we improve the perceptual quality of that part of the video while on the other hand, if this technique does not find enough number of edges in a particular part of the video, it is presumed that that part of the video perhaps would not be able to get the as much user's attention as the part of the video having edges. Therefore, we introduce a deliberate distortion that helps to reduce the overall bitrate. It also reduces the complexity of the encoder.

4.4.2.4 Results and Discussion

In this section, we first describe some preliminary results supporting the validity of our proposed scheme in terms of bitrate reduction at the client end for receiving the wireless video data. We also analyze reasons for the marginal loss in PSNR. An important point to note here is that our proposed encoding technique is for real time encoding unlike [9] which performs the encoding offline on a video file.

The proposed rate control algorithm is implemented in the H.264 JM10.2 baseline profile and is compared with the rate control algorithm in JM10.2. The same encoding parameters were used for both algorithms in order to ensure the comparison is fair. We were able to reduce the bit rate but introduced a small loss in PSNR. The maximum PSNR loss suffered in this case is about 0.81 dB. In some cases PSNR was gained along with the reduction in bit rate. Figure 4.4.2.2 shows the bit rate gain for 4 QCIF sequences. It was also noted that this methodology worked well at low bit rate from 32 kbps to 256 kbps which is the typical bandwidth in wireless devices. The JM 10.2 software used in the evaluation is designed for algorithmic development and is not a good indicator of performance gains in real H.264 encoder products. We are currently implementing the proposed method in other available implementations of H.264 standard implementations.

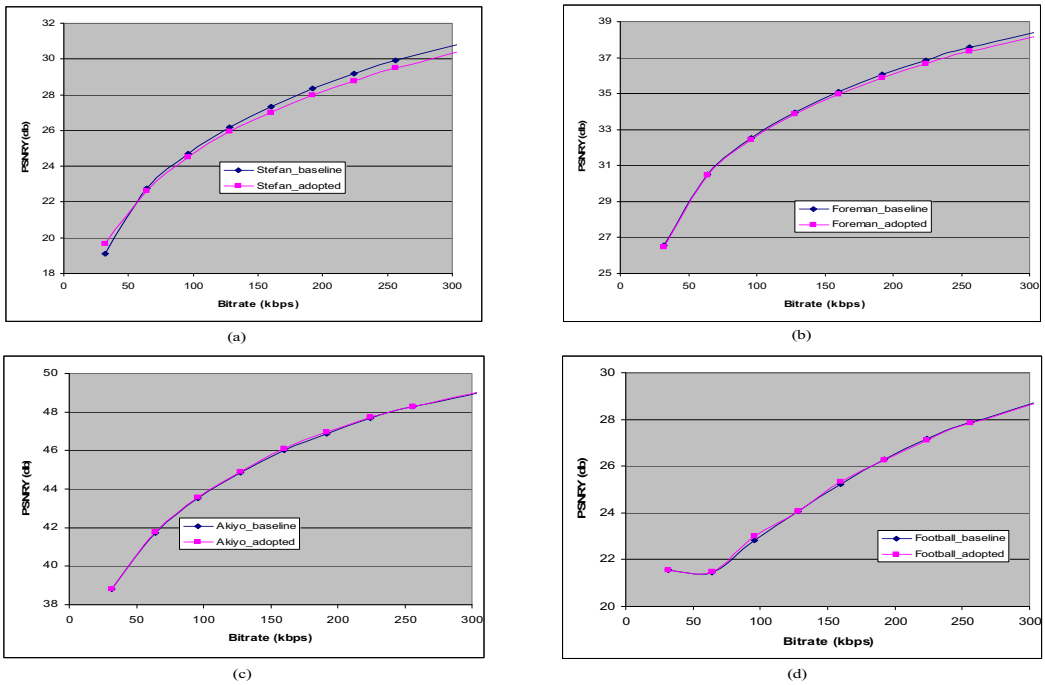


Figure 4.4.2.2. PSNR loss and Bitrate gain for QCIF sequences: (a) Stefan.qcif, (b) Foreman.qcif, (c) Akiyo.qcif, and (d) Football.qcif.

Another observation about the results is the trend of bitrate gain is more among the sequences having more complex regions or contents e.g. the sequences having movements in large portion of the video and sequences having scene changing in them tend to reduce more bitrate than those sequences which don't have a lot of movement or dull contents in them. The former sequences include Mobile, Coastguard and Stefan while later sequences include Foreman, Akiyo and Football.

Figure 4.4.2.3. Perceptual improvements in Foreman sequence.

Figure 4.4.2.3 shows the comparison of perceptual quality of a frame that is encoded (a) without any change in the encoder (b) with our proposed technique. This comparison clearly shows that the image contents seem to having some kind of

activity are identified by our algorithm as having edges while the other parts of the image are identified as dull contents i.e. no edges detected. Therefore, visually important regions of this frame have better quality than the remaining regions. Thus the perceptual quality is visibly enhanced in this video frame which we obtained by adaptive quantization as part of our technique.

4.4.2.5 Conclusions

In this paper we presented a technique to enhance the perceptual quality of H.264 video for mobile applications by using adaptive quantization based approach applied to the H.264 baseline profile. The proposed approach has great potential to improve quality of the visually important regions in a video as well as to reduce the computational complexity by reducing the bitrate. The results show that the RD performance is very close to the reference encoder. The encoding time is often reduced compared to the reference encoder.

Our technique is based on the fundamental observation that contents of the video having edges attract the viewer's attention more than the other parts of the video. Our technique can be integrated seamlessly into existing H.264 codecs which can have any rate control algorithm implemented in them. Since it does not require additional parameters to be passed during the streaming process to the decoder, it is easy to use with existing H.264 codecs. It can also be used in real-time encoder instead of using it offline on a video file. We focus on, but not limited to, the rate control by adaptive quantization and investigating on further techniques in rate control extensions. We demonstrated by comparison that appropriate rate control techniques lead to successful implementations with superior performances.

References for Section 4.4.2

[1] H. Kalva, "The H.264/AVC Video Coding Standard," IEEE Multimedia, Vol. 13, No 4, Oct.-Dec. 2006, pp. 86-90.
[2] A. Luthra, R. Gandhi, K. Mckeon, Y. Yu, K. Panusopone, D. Baylon, L. Wang. "Performance of MPEG-4 Profiles used for Streaming Video and Comparison with H.26L". M7227, Motorola, Broadband Communications Sector, July 2001.
[3] Z. Li, F. Pan, K. –P. Ling, G. Feng, X. Lin, S. Rahardja, "Adaptive basic unit layer rate control for JVT", JVT-G012, March 2003.
[4] G. Sullivan, T. Wiegand, K.-P. Lim, "Joint Model Reference Encoding Methods and Decoding Concealment Methods", JVT-I409, September 2003.
[5] T. Wiegand, H. Schwartz, A. Joch, F. Kossentini, G.J. Sullivan, "Rate Constrained Coder Control and Comparison of Video Coding Standards", IEEE Trans. Circuits Systems Video Technol. 13 (2003) 688-703.
[6] G. Fernandez-Escribano, H. Kalva, P. Cuenca, and L. Orozco-Barbosa, "Very Low Complexity MPEG-2 to H.264 Transcoding Using Machine Learning," Proceedings of the ACM Multimedia 2007, Santa Barbara, CA, October 2006, pp. 931-940.
[7] D. Wu, T. Hou, Y. –Q. Zhang, "Transporting real time video over the Internet: challenges and approaches", Proc. IEEE 88 (2000) 1855-1877.
[8] Z. Ni, Z. Chen, K.N. Ngan, "A real-time video transport system for the best effort Internet", Signal Processing: Image Communications 20 (2005) 277-293.

[9] Siddhartha Chattopadhyay, Suchendra M. Bhandarkar, Kang Li, "FGS-MR: MPEG4 Fine Grained Scalable Multi Resolution Layered Video Encoding", Proc. Of ACM NOSSDAV'06, New Port, Rhode Island, May 22-23, 2006 pp.
[10] Schaar, M. van der, and Lin, Y.-T., "Content-based selective enhancement for streaming video", in Proc. IEEE International Conference on Image Processing, Vol. 2, pp. 977-980, 2001.
[11] J. B. Lee and A. Eleftheriadis, "Spatio-temporal model-assisted compatible coding for low and very low bit rate video-telephony," Proc. IEEE Int. Conf. Image Processing Lausanne, Switzerland, pp. 429-432, Oct. 1996.

4.4.3 3D Video Compression

This paper presents the results of the 3DTV quality evaluation on autostereoscopic displays using asymmetric view coding. Asymmetric view coding encodes the stereo views with different quality. It has been shown that the human visual system is able to compensate for this asymmetric view quality and present a good quality 3D video. Asymmetric video coding can be exploited to reduce the bandwidth requirements for 3DTV services. The key factors that affect the asymmetric video coding are the compression algorithms, the human visual system, and the 3D display. We conducted a subjective evaluation of 3D video with asymmetric view quality and encoded using MPEG-2. We also studied the impact of eye dominance on the perceived quality. We show that asymmetric view coding can be used to reduce the bandwidth requirements of 3DTV services based on MPEG-2 view coding.

4.4.3.1 Introduction

3DTV services are beginning to receive significant attention from researchers and the industry. The advance in video coding together with the advances in 3D displays will allow easier deployment of 3DTV services. One of the keys to 3D TV services is the ease of use and viewing comfort [1]. Autostereoscopic displays are emerging as good candidates for 3DTV. Autostereoscopic displays do not require 3D viewing glasses and the most common ones based on lenticular imaging technologies are relatively inexpensive to manufacture. The quality, cost, and success of 3D video services deployed will depend on the technologies used for compression, communication, and display.

3D video perception requires a pair of views, the left view and the right view, to be presented to the left eye and the right eye of users. The two views can be coded independently requiring twice the bandwidth of the traditional TV or interview coding techniques that allow prediction across view can be employed to improve compression efficiency. The bitrate required for 3D video services can be substantially reduced if the human visual system is properly exploited. Human visual system has the remarkable ability to compensate for the loss of information in one of the views and still present a very good 3D video perception. This is essentially the case where a person with perfect vision in one eye and a slightly blurry vision in the other eye is able to see the world around him normally. This ability of the human visual system can be exploited to reduce the compression of 3D video services by applying asymmetric view coding. In asymmetric view coding the left and the right eye views are encoded with different qualities without degrading the 3D experience.

A study on the bounds of asymmetric view coding using H.264 was presented in [2]. A 3DTV service offered using asymmetric video coding allows multiple services with minimal bandwidth requirements. For example, the high quality view can be used for the traditional TV and the additional low quality view can be delivered only to the users with a 3DTV. This approach allows for gradual deployment of 3DTV services.

Given that MPEG-2 hardware is used virtually in all digital TV services, a 3D video based on MPEG-2 video coding should be evaluated. The more recent H.264 video is another candidate for 3DTV services. While MPEG-2 video takes significantly more bandwidth compared to H.264, the existing MPEG-2 infrastructure could be leveraged in 3D TV services. The lower bitrates required by H.264 reduces the bandwidth requirements for 3DTV services but requires new hardware deployments. Since H.264 is not expected to replace MPEG-2 in the short term, an MPEG-2 based service with asymmetric view coding offers service providers a 3DTV service with lower deployment costs. The quality of 3D video depends on the coding algorithm, 3D display used, and the human visual system. In this paper we present a quality evaluation and study the role of eye dominance on the 3D video using asymmetric view coding based on MPEG-2. The goal of this work is to understand the impact of the coding algorithms and human visual system on the perceived quality of the 3D video coded using asymmetric view coding.

The rest of the paper is organized as follows. Section 2 gives a brief overview of 3D perception, experimental methodology is presented in Section 3, the results are discussed in section 4 and conclusions presented in Section 5.

4.4.3.2 3D Perception

The human visual system receives two separate projections of a scene; one from each eye. Human eyes are separated by an average horizontal distance of 2.5 inches and the images captured by the eyes are slightly different. The left and right eye views are combined in the brain resulting in a single 3D percept. The combined visual perception of the scene is also known as binocular fusion. Binocular suppression is property where portions of the view in one eye are suppressed by the corresponding view of the other eye. The possibilities of dominance and suppression mechanisms during the binocular fusion exist, but their impact is not yet well understood [3]. Experiments have shown that when the left and right eye views are combined the higher quality view is able to mask coding artifacts in the lower quality view [4, 5].

The process of binocular fusion in the human visual system results in the comparison and combination of the left and right eye views to generate a single 3D percept. The left and right eye views have to be presented to the users using 3D display means to give the sensation of 3D and depth perception. The left and right eye views can be encoded and sent to the receiver and the stereo views can be generated at the receiver. The properties of binocular fusion make possible encoding of left and right eye views at different bitrates. This asymmetric view coding has been exploited to improve compression efficiency [4, 5]. Asymmetric view coding for 3D TV based on H.264 was reported in our prior work [2]. Since the compression artifacts influence 3D perception, the effect of MPEG-2 coding artifacts is expected to the different compared with the H.264 coding artifacts. The past studies using MPEG-2 based 3D coding have not studied the impact of eye dominance on the perceived quality. The

effects of the eye dominance and autostereoscopic displays on the 3D video quality cannot be understood from the past MPEG-2 based studies.

The two main approaches for delivering 3D video are 1) stereo coding where the left and right views are encoded and 2) depth image based rendering (DIBR) where a single view and an associated depth map are transmitted to the receiver. DIBR systems synthesize the left and right views at the receiver based on the single view and the depth information. We evaluate 3D TV based on stereo view coding in this work.

4.4.3.3 Experimental Methodology

The goal of this work is to understand the impact of the eye dominance and autostereoscopic displays on the quality of the 3D video experiences. We are currently conducting a large user study to evaluate the impact of asymmetrically coded 3D views on the quality of the 3D video rendered on the Sharp autostereoscopic display. The goal of this study is to understand the bounds of asymmetric coding, relationship between the eye-dominance and 3D quality of asymmetrically coded video. The results are reported based on the evaluations from 20 users that have predicated in the study.

The sequences used for these experiments are the Akko & Kayo and the Ballroom sequences created for 3D/mulitview coding work currently underway in the MPEG committee [6]. A pair of views from these sequences was chosen to render stereo video. The video sources are 10 seconds long, 640x480 resolution, 30 FPS, and available in YUV 4:2:0 format. The Akko & Kayo sequence is made specifically for this research and has a number of carefully selected objects that help evaluation of 3D sequences well. The Ballroom sequences capture ballroom dancing and show dancers at multiple levels of depth.

The test sequences were created to test 3D video using asymmetric view coding and coded at different levels of quality. The quality was varied by encoding the left and right eye views at different qualities. Two test cases were created for each video sequence: 1) right eye view at a high quality with left eye view quality varying and 2) left eye view kept constant at a high quality and the right eye view quality varying. The high quality view that was kept constant was encoded with a PSNR of about 42.7 dB for the Akko & Kayo sequence, and the quality of the other view is varied from 42.7 dB to 33 dB. For the Ballroom sequence the high quality view that was kept constant was encoded with a PSNR of about 39.6 dB, and the quality of the other view is varied from 39.6 dB to 30 dB. In both cases, two adjacent views of the multi-view sequences were used as a stereo pair.

The subjective evaluation was done by 20 participants who evaluated the overall quality of video (without looking for specific artifacts) on the standard subjective evaluation scale from 1 (unacceptable) to 5 (excellent). For each user a test was performed to identify the dominant eye and the eyedness (left or right eye dominant) and handedness (left or right handed) information was noted. The study had 12 right eye dominant and 8 left eye dominant participants; 16 were right handed and 4 were left handed. The participants were presented with 10 second test videos separated by a 5 second mid-gray screen. The left and right eye views were encoded with quality

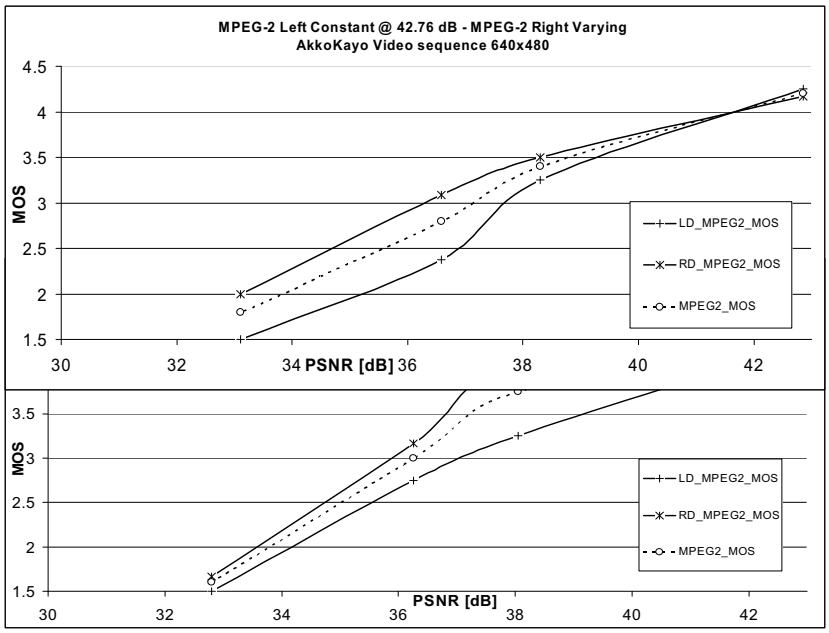
varying from high quality to low quality and stereo pairs were created with a high quality view and a lower quality view thereby creating a 3D video with asymmetric view quality. Each video presented was randomly selected from the test set.
We used the Sharp LL-151-3D autostereoscopic display and a Sharp 3D laptop to evaluate the 3D quality. Both the display is 15-inches, XGA resolution (1024 by 768 pixels). This display which uses lenticular imaging techniques and renders depth very accurately gives a true 3D experience. The perception of depth is achieved by a parallax barrier that diverts different patterns of light to the left and right eye.

4.4.3.4 Results and Discussion

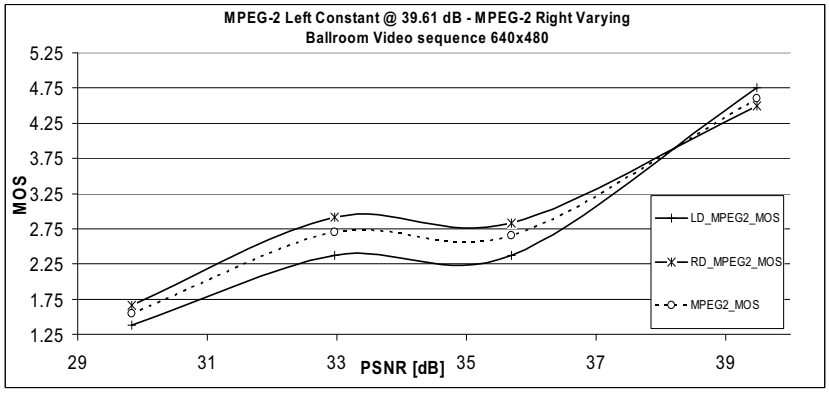
The quality of a single 2D view alone is not an indication of the 3D quality. The PSNR of the individual views can be used as a metric when the quality of both views is high. With asymmetric view coding quality is influenced by the coding artifacts present in the individual views, the human visual system, and the type of 3D display. Developing objective quality metrics for 3D quality is thus very difficult and subjective evaluation is the primary means of evaluating 3D video quality. We evaluated the role of the human visual system by studying the role of eye dominance.

Humans have a preference for one eye over the other, referred to as eye dominance. The significance of eye dominance, however, is not well understood. Humans are mostly right handed (90%) and about 70% are right eyed, 20% left eyed, and 10% exhibit no eye preference [7]. A recent study suggested that the eye dominance just indicates individual sighting preferences and has no function in binocular vision [8]. Another study found that eye dominance improves the performance of visual search tasks by perhaps aiding visual perception in binocular vision [9]. Our results also suggest a role for eye dominance in 3D perception when asymmetric coding is used.

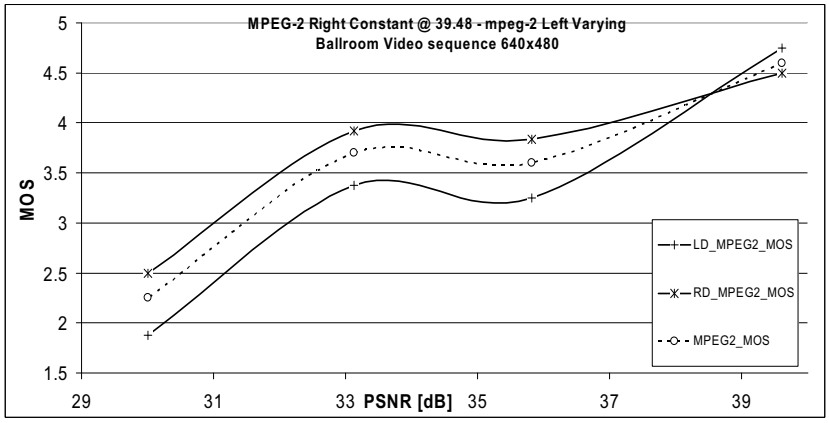
Mean opinion scores were computed for the test sequences based on subjective evaluations. Figure 4.4.3.1 shows the mean opinion scores (MOS) for the Akko & Kayo and Ballroom sequences. Figures 1.a and 1.c show the MOS with left eye view coded at a high quality while the quality of right eye view in each evaluated sequence was varied from high to low. Figures 1.b and 1.d show the MOS with right eye view coded at a high quality while the quality of left eye view was varied. Figure 4.4.3.2 shows the frame 100 of the two views the Ballroom sequence coded with asymmetric quality. As the quality of a view drops, the MOS of the perceived 3D quality also drops. The rate of drop in MOS is different for the Akko & Kayo and Ballroom sequences, The Akko & Kayo sequence is slow motion sequence and artifacts are more easily detected. The Ballroom sequence has lot of motion with dancers moving across the screen and the drop in perceived 3D quality is not as steep. In both cases the perceived quality drops faster as the asymmetry in the coded view quality increases.



(b)



(c)



(d)

Figure 4.4.3.1. Mean Opinion Score of subjective evaluation tests for asymmetric view coding.

The plots also show the influence of eye dominance. The figures also show the MOS of users grouped by their eye dominance. The perceived 3D video quality is clearly influenced by the eye dominance. The role of eye dominance seems to increase as the asymmetry in the coded view quality increases. The plots show that the right eye dominant users perceive better quality irrespective of the quality of the right eye view. The left eye dominant users are more sensitive to the view quality asymmetry.

Figure 4.4.3.2. Stereo views with asymmetric view quality. Top: right view coded at 6 Mbps (39.6 dB PSNR); Bottom: left view coded at 1 Mbps (30 dB PSNR).

4.4.3.5 Conclusion

This paper evaluates the use of asymmetric view coding for 3D video services. With asymmetric view coding, quality is influenced by the coding artifacts present in the individual views, the human visual system, and the type of 3D display. The wide use of MPEG-2 infrastructure in digital TV services means a careful study of 3D TV using MPEG-2 is required. Though MPEG-2 has a higher bandwidth requirements compared to H.264, the use of asymmetric view coding can reduce the overall bandwidth required for 3D video services. Services employing asymmetric view coding have to be carefully deployed as the perceived quality is influenced by the eye

dominance of the end users. The right eye dominant uses seem to perceive better 3D quality and less sensitive to view quality asymmetric compared to the left eye dominant users. Since about 20% of the general population is left eye dominant, the stereo views cannot have large asymmetry without affecting the perceived quality of the end users. Asymmetric view coding presents an interesting option for reducing the bandwidth requirements of 3D video services. The asymmetric view coding can be exploited further if content based encoding is employed. The lower quality view can be coded such that the visual cues that contribute to 3D perception are coded with a higher quality compared with the regions without any depth cues.

References for Section 4.4.3

[1] H. Kalva, L. Christodoulou, L. Mayron, O. Marques, and B. Furht, "Challenges and opportunities in video coding for 3D TV," IEEE International Conference on Multimedia & Expo (ICME) 2006, July 9-12, 2006, Toronto, Canada.
[2] H. Kalva, L. Christodoulou, L. Mayron, O. Marques, and B. Furht, "Design and Evaluation of 3D Video System Based on H.264 View Coding," International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV 2006), Newport, Rhode Island, May 22–23, 2006, pp. 68-73.
[3] O. Schreer, P. Kauff, and T. Sikora, edts., "3D Video Communications" Wiley 2005.
[4] Lew B. Stelmach, W. James Tam, "Stereoscopic image coding: Effect of disparate image-quality in left- and right-eye views", Signal Processing: Image Communication, Vol. 14, pp.111-117, 1998.
[5] Daniel V. Meegan, Lew B. Stelmach, and W. James Tam, "Unequal Weighting of Monocular Inputs in Binocular Combination: Implications for the Compression of Stereoscopic Imagery", Journal of Experimental Psychology: Applied, Vol. 7(2) 143-153, Jun 2001.
[6] ISO/IEC JTC1/SC29/WG11, "Call for Proposals on Multi-view Video Coding (MVC)," MPEG Document MPEG2005/N7327, July 2005.
[7] D.C. Bourassa, I.C. McManus, and M.P. Bryden, "Handedness and eye-dominance: A meta-analysis of their relationship," Laterality, Vol 1, No. 1, 1996, pp. 5–34.
[8] A.P. Mapp, H. Oho, and R. Barbeito, "What does the dominant eye dominate? A brief and somewhat contentious review," Perception & Psychophysics, Vol. 65, No. 2, 2003, pp. 310– 317.
[9] E. Shneor and S. Hochstein, "Effects of eye dominance in visual perception," International Congress Series, Volume 1282, Vision 2005, September 2005, pp. 719-723.

4.4.3.6 3D Encode

This work is related with the depth encoding for 3d video sequences available for Phillips monitor.

4.4.3.7 Methodology

1. Separate 3D videos in 2D frame and depth frame, take depth frame and create a depth sequence YUV, having in mind we only have Y component. Then we

will make h.264-JM12x (it is for Y only) encoding with different parameters, the depth sequences we have.
2. Calculate PSNR of encode depth sequences, based in the best quality possible.
3. Compose the 3D video with encode depth sequences, and evaluate (using people) the quality of the result 3D video.

We were dealing with the WMV to YUV problem, it was difficult to find codec's for them but finally we are able to read s3d files (original 3d videos for Phillips monitor), changing the extension to wma.

We used the Mplayer (linux player) to get the YUV sequences, the YUV sequences are in YUV4Mpeg format which is basically a YUV with headers.

4.4.3.8 Tools

YUVh2YUV : read YUV4Mpeg format, removes headers and writes the result in a standard YUV format
YUVh2Depth: read yuv4mpeg format, get Depth information and creates standard YUV with depth information
YUVh22Dinfo: read yuv4mpeg format, get 2D information and creates standard YUV with 2D information
Compose3D: Compose again 2D information and Depth information into a standard YUV format.

To be able to watch 3d video we convert YUV to AVI (yuv2avi) the program already exists

4.4.3.9 Coding and decoding

We are using the IPP tool for encoding. We compare the "original depth video" against different encodings (PSNR).

The proposed resolutions go from 100 kbps to 1mbps, with step of 100kbps.We also designed an experiment for the quality results in the 3d monitor. We selected 4 videos to make the test. The tests are going to be run from the MLAB server.

We had to resize the files to 300 frames because the larger file crashes the system and take a long time. We adjust the programs YUVh2Depth and YUVh22Dinfo, to have the option of choose skip frames and total frames. Six shots from different sequences were selected for the tests; finally the programs were modified in order to selects the initial and final frames. The process (the storage of the huge YUV files) wasn't made in the MLAB server because the transfer makes it slow and corrupts the process. For the six shots we separate the 2d info and depth info, the depth was encoded in h264 (100k to 1000k)

The files were put together using compose3d.exe then create an avi file with yuv2avi.exe and finally we used windows media encoder to create .wmv files. (The configuration for windows media encoder is in 3dencode.wme in the same folder). It has been used 3 computers for this during 12 hours.

Also we created a excel file with the PSNR results.

Sequence	kbps	shot1 PSNR	shot2 PSNR	shot3 PSNR	shot4 PSNR	shot5 PSNR	shot6 PSNR
	100	29.90406	30.24847	32.24656	32.43419	33.7442	41.22857
	200	32.57488	32.88103	36.00944	35.92042	37.56456	43.83509
	300	34.51115	35.85183	37.11329	39.73543	40.07597	46.32143
	400	36.34594	36.4895	38.1863	41.40854	43.2114	49.03419
	500	38.57026	38.27285	38.95314	43.30166	44.53178	50.5369
	600	37.95281	38.94581	40.10017	44.68956	44.53028	51.20998
	700	41.10178	39.42503	40.66855	45.05131	46.14626	52.16247
	800	43.14694	40.60144	42.46352	45.62881	46.39031	52.64594
	900	44.21471	42.04474	42.9376	46.2821	46.99156	52.83443
	1000	45.05489	42.18853	44.20198	46.71099	46.88252	53.44806

Now we have to design an experiment for human evaluation of the quality of the encode, we have 6*10 videos.
Description of the experiment to evaluate the quality of the encoded depth info:
For the test we select 4 bit rates 100k, 200k, 500k, and 1000k
The experiment will be the following.
We need to determine which eye is the dominant in each person, for this we are using:
The "Dolman method" for determination of ocular dominance also known as the "hole-in-the-card test"
The subject is given a card with a small hole in the middle, instructed to hold it with both hands, then instructed to view a distant object through the hole with both eyes open. The observer then alternates closing the eyes or slowly draws the opening back to the head to determine which eye is viewing the object (i.e. the dominant eye).
Then sort the selected sequences in random order and use Mean Opinion Score (MOS) like this:

MOS	Quality
5	Excellent
4	Good
3	Fair
2	Poor
1	Bad

We pretend to evaluate so many people as possible.
1. We need to create the card for the ocular dominance test.
2. Create the sequences for the experiment.

A	shot6300k.wmv
B	shot2300k.wmv
C	shot1300k.wmv
D	shot61000k.wmv
E	shot1500k.wmv
F	shot51000k.wmv
G	shot6100k.wmv

H	shot6500k.wmv
I	shot4500k.wmv
J	shot5500k.wmv
K	shot1100k.wmv
L	shot21000k.wmv
M	shot4100k.wmv
N	shot5300k.wmv
O	shot31000k.wmv
P	shot2100k.wmv
Q	shot11000k.wmv
R	Shot3300k.wmv
S	Shot41000k.wmv
T	Shot3500k.wmv
U	Shot2500k.wmv
V	Shot5100k.wmv
W	Shot3100k.wmv
X	Shot4300k.wmv

At the moment that this report is being made, we have had 20 persons evaluated. we are expecting to evaluate more people.

4.4.4 Multi-view Video Navigation Using Motion Sensing Remote Controllers

The goal of Multi–view Video Coding is to allow coding of multiple camera views such that the user has the freedom of choosing the view point. While there have been significant efforts in multi-view compression, the work on navigation has been limited. The biggest challenge here is in developing user friendly and intuitive navigation of multi-view content. We propose a system which consists of multi-view video player and a motion sensing remote controller. Motion sensing remote controllers became widely available with the introduction of the Nintendo Wii game console. The remote controller captures motion when waved and the motion information is used to change the views in multi view video applications. This system gives users a convenient way of navigating through the multiple views by waving the remote in the desired direction.

4.4.4.1 Introduction

Video coding technologies have matured sufficiently to make possible new generation of video applications. Multi-view video coding (MVC) has begun receiving significant attention in the industry. The goal of MVC is to allow coding of multiple camera views such that the user has the freedom of choosing the view point. In MVC, multiple cameras are used at the sender to capture the same scene and a multi-view video encoder is used to compress the output off all the cameras jointly. The compressed stream is delivered to a receiver over networks. The receiver can be a TV without multi-view capability, a 3D TV system, or a multi-view receiver with interactive view selection. Research on multi-view coding has gained attention over

the last two years. An overview of MVC technologies submitted to the MPEG committee as a part of the standardization process is presented in [1]. The existing technologies that can be applied to multi-view coding are reported in [2,3].

Fig. 4.4.4.1 (a)

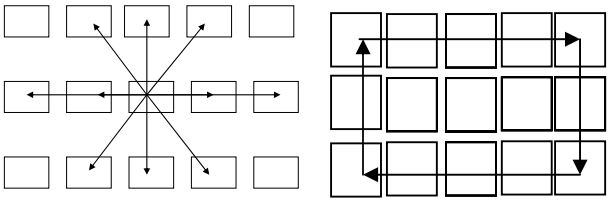
Figure 4.4.4.1 (b)

Figure 4.4.4.1. Two common arrangement of cameras in multi-view video applications; figure 4.4.4.1.a shows five-view video and figure 4.4.4.1.b shows an example of 15-view content. A 1D arrangement has cameras placed in a single row or single column and 2D camera arrangement has cameras in both rows and columns. Since the multiple views are very similar, users are expected change views to get more detail in certain spatial direction. In such applications, playing a specific view may not have any more significance than playing another view that is close. These characteristics of multi-view video applications necessitate novel ways of navigating and playing multi-view video. The novel contribution of this paper is an innovative approach to multi-view video navigation using motion sensing remote controllers.

4.4.4.2 Background and Motivation

There is limited or no work on navigation in multi-view video applications. With multiple views available, the key question is how to present the right view to a user? Or, how can a user select a view? Automating view selection without user intervention can be done by providing view change hints in coded multi-view bitstreams. This however, may not be sufficient as users desire control over the view displayed. With the number of available views known to the users, the users can enter the view number on a remote control or the receiver to change the view. This solution is disruptive to the user experience as the users have to think and decide the view to be displayed. Gaze estimation and changing a view based on a user's gaze is a possibility. A gaze based system for stereo video is reported in [4]. Gaze has limited value in practical multi-view TV applications as TV programs can be watched by more than one user and users' gaze is likely to shift without the users intending to change the views. A remote control based approach to selecting stereo views is reported in [5] and a similar approach can be used for multi-view video. This approach is also limited due to the fact that users have to explicitly select the views by visually selecting the views on a remote controller.

The recent introduction of motion sensing remote control by Nintendo has enabled new possibilities for user interaction with content. We propose to use such a motion sensing remote control to create a natural way for user interaction and navigation in multi-view video applications. By motioning the remote control in the direction of view the view can be changed. The users can always fall back on the traditional method of inputting the view number if necessary. The typical directions of motion supported by motion sensing remote controllers are shown in figure 4.4.4.2. Figures shows the motion directions overlaid on a multi-view schematic with 15 views.



(a) Translation (b) Rotation

Figure 4.4.4.2. Motion based view selection.

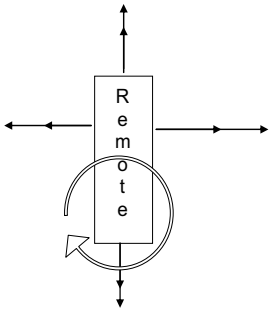


Figure 4.4.4.3. Remote control motion.

4.4.4.3 System Development

Figure 4.4.4.3 shows the possible remote control motion. For multi-view navigation, the motion has to be translated into view selection. Since the range and speed of remote control motion are different for individuals, the motion has to be calibrated for each user. The calibration step consists of users performing full range motion at normal speed. The four types of motion calibrated are: vertical, horizontal, diagonal, and rotation. As the number of views and view layout can vary for each presentation, the motion has to be mapped to the view layout of the current video. When users move the remote for changing the view, the amount and type of motion is recorded and then the view selected is determined based on the initial calibration.

The remote control motion on successive movements can be combined to create more complex navigation operations. A full range motion to the left followed immediately by a full range motion to the right, for example, creates cyclic motion through all the horizontal views of that row. A set of motion combinations are developed to perform complex navigation operations. The motion based view selection is supplemented by providing navigation configuration settings that control the navigation. The settings provided are: 1) maximum numbers of linear views changed 2) cycling speed for view rotation 3) enable motion combinations 4) auto calibration.

Implementation Using Nintendo Wii Remote Controller

The Nintendo Wii remote controller (Wiimote), has been widely available since the introduction of the Nintendo Wii game console in December 2006. The key components of the Wiimote that enable the motion sensing controller are: Bluetooth receiver and transmitter, accelerometer, IR sensors, and tilt sensors. A Wiimote is connected to a receiver/computer using the Bluetooth Human Interface Device (HID) [6]. HID is similar to the USB HID and implementation in a device is mostly the

same. A few open source projects provide APIs to access the Wiimote protocol and motion data [7].

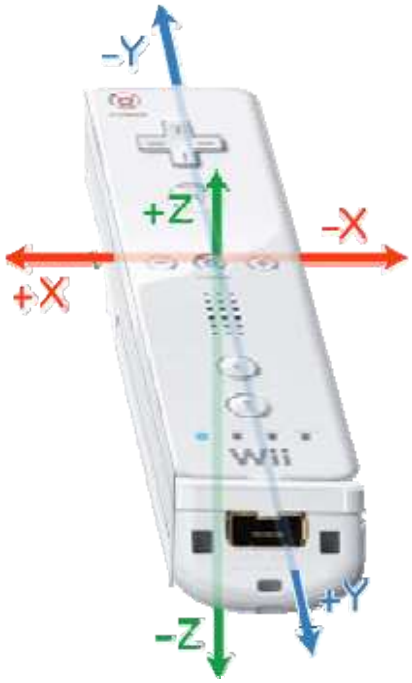


Figure 4.4.4.4. The Accelermoter Axis from Wii Linux Wiki.[7]

The accelerometers on a Wiimote measure the X, Y and Z movement. With accelerometers, one is able to tell the direction and speed of the Wiimote motion. The accelerometer used inside the Wiimote is the ADXL330[9]. This chip uses Gs as a measure of acceleration. This chips gives us a range of 3 Gs in both direction. The calibration of the Wiimote to an user to change view can be difficult. The program has to be able to adjust the sensitivity to an user and also the force which causes a view change. The two axes which have to be taken into consideration is the x and z. The z axis is tricky because the accelerometer picks up gravity as an acceleration force. To over come this, the program is adjusted to expect greater force for the up and down motion. A Wiimote comes with a IR sensor bar that is placed in front of the user and near the screen. The IR sensor bar is composed of two groups of IR LEDs that are used to determine the relative position of the Wiimote. The motion information is communicated to the receiver over the Bluetooth connection. The motion sensors on the Wiimote are not as accurate as one would hope. Improvement with the data output has been shown using Kalman filters [8]. The calibration stage maps the motion range to the view range and the view selected is determined based on the motion detected.

The Wiimote buttons are used in addition to the motion information to provide control and navigation functions. We use the trigger button at the front-bottom of the remote to pause time and the remote motion causes the change in view resulting in temporal pause and spatial motion. The + control pad is used to precisely move one view at a time. This feature can be used as a means of refining the currently selected view after view selection based on motion. The home button is used to return to the default or

anchor view. The VCR-like play control actions are mapped to other buttons on the controller and using motion combinations. The A-button is mapped to play-pause, minus button (-) is mapped to rewind, and plus button (+) is mapped to fast forward. The button mapping can also be changed using the controller settings menu on the host PC. The Wiimote motion together with buttons gives a flexible and user friendly controller for multiview video navigation.

References for Section 4.4.4

[1] ISO/IEC JTC1/SC29/WG11, "Survey of Algorithms used for Multi-view Video Coding (MVC)," MPEG Document MPEG2005/N6909, January 2005.
[2] T. Matsuyama, "Exploitation of 3D Video Technologies," Proceedings of ICKS 2004, pp. 7-14.
[3] A. Smolic, K. Mueller, P. Merkle, T. Rein, M. Kautzner, P. Eisert, and T. Wiegand, "Representation, Coding, and Rendering of 3D Video Objects with MPEG-4 and H.264/AVC," 2004, pp. 379-382.
[4] Y.-M. Kwon et. al., "3D Gaze Estimation and Interaction to Stereo Display," The International Journal of Virtual Reality, 2006, 5(3):41-45 41.
[5] M.-C. Park, S. K. Kim, and J.-Y. Son, "3D TV Interface by an Intelligent Remote Controller," Proceedings of the 3DTV Conference Kos, Greece, May 2007
[6] C. Ranta and S. McGowan, "Bluethooth HID Spec," HTTP: http://www.bluetooth.com/NR/rdonlyres/0BE438ED-DC1B-41D1-AAC0-1AAA956097A2/980/HID_SPEC_V10.pdf
[7] Wii Linux Wiki , "WiiMote," [Online document], 2007 June 6,[cited 2007 June 28], Available HTTP: http://www.wiili.org/Wiimote
[8] B. Rasco, "Where's the Wiimote? Using Kalman Filtering To Extract Accelerometer Data," http://www.gamasutra.com/view/feature/1494/wheres_the_wiimote_using_kalman_.php

4.5 Algorithms for Detection and Tracking of Video Objects in Single-View and Stereo, and Survey Study of Suitability of Several Image Databases for Attention-based Image Classification, Retrieval, and Detection of Regions of Interest

This part of the report describes a novel method for object detection and tracking based on the use of neural-network segmentation and MPEG-7-like descriptors from stereo sequences. It also describes techniques for detecting humans and classifying tracked objects. A novel method based on use of optical-flow for video object segmentation and tracking is proposed. Finally, we present an survey of video databases and their suitability for several biologically-inspired attention-driven methods for classifying and retrieving digital images.

4.5.1 Robust Detection and Tracking of Video Objects in Stereo for Smart Video Surveillance

This section presents a design of a system for video surveillance employing object detection and tracking which integrates depth information from a pair of cameras. It is a part of a smart maritime video surveillance system in which robustness and near real-time processing are among the major design goals. A robust surveillance system

must aim to produce a minimal amount of false positive results while simultaneously keeping the number of false negatives as low as possible. Furthermore, such a system must be able to track both rigid and non-rigid objects in complex environments and overcome automated tracking difficulties that arise due to object occlusion. Unlike many other surveillance systems, our system uses stereo video footage to estimate depth information, improving the quality of object detection and tracking. The method consists of object segmentation based on a novel class of Bayesian probabilistic neural networks, computation of a depth map, and object tracking based on feature descriptors including intensity, color, shape, motion and depth. Experimental results are provided to demonstrate the performance of our approach.

4.5.1.1 Introduction

Among the challenges facing the design of a truly automated system for surveillance is ensuring that it is capable of fast and accurate object detection, object tracking, object identification, and object interpretation. The volume of surveillance data is typically too much for an unassisted human operator to effectively attend to, hence the need for the development of automated solutions. However, the effectiveness of object identification and interpretation is dependent on the quality of tracking data it receives, which in turn depends on the quality of detected objects. An object that is never detected cannot be tracked nor identified. Likewise, a falsely detected object or incorrectly tracked trajectory wasted processing resources and reduces the performance of identification and interpretation modules. We present a new approach to the critical object detection and object tracking modules which is enhanced by stereo visual data in the context of an automated digital video surveillance system.

At a high level, smart automated surveillance systems generally consist of two task-driven modules. The first is a module that automatically detects and tracks the objects of interest (e.g. humans or vehicles) from an incoming video stream. This can be accomplished with background subtraction or segmentation (for object detection) and a variety of tracking techniques. The second module recognizes and identifies the detected and tracked objects and classifies their behavior (either as normal/benign or abnormal/alarming). While certain tasks can be accomplished in either or both modules (such as eliminating false positives), the output of the tracking and detection module is a prerequisite of the identification and interpretation module. We describe an approach to the first module, object detection and the subsequent tracking. Our ultimate objective is the design a module that is robust to object shape, complex or moving backgrounds, and full and partial object occlusion, which is also fast and suitable for real-time monitoring and surveillance applications.

We use an approach based on a novel class of Bayesian neural networks for robust object detection. This approach is capable of successfully segmenting foreground objects in digital video. This method is adapted for object detection and tracking. A fast method of rough depth estimation from stereo video sequences is employed to improve the handling of object occlusion as well as to learn relative distances between objects (given an a priori reference), enhancing the basic segmentation results. To track each object a set of feature descriptors, including intensity, color, shape, motion, and depth features, is generated for all foreground objects. The work falls within the context of our target of the creation of a complete system for video surveillance along the lines of VSAM [1] and W4 [2].

4.5.1.2 The Proposed Framework

Fig. 4.5.1.1 presents the proposed framework. It consists of the following major components:

Video capture: in our implementation digital video is acquired from two cameras mounted in a stereo configuration (a left camera and a right camera);

Object detection: this is accomplished by using background modeling neural networks;

Depth estimation: a fast, real-time method for determining depth on an object level is employed;

Feature extraction: both localized features of video objects (including object's size, position and depth) as well as their global appearance features (including color, shape and texture) are extracted.

Feature update and comparison: extracted features for objects in the current frame are compared to previously extracted features (for the same objects as well as other objects) and adapted according to measures of their discriminative power;

Object tracking: point-based object tracking has been implemented.

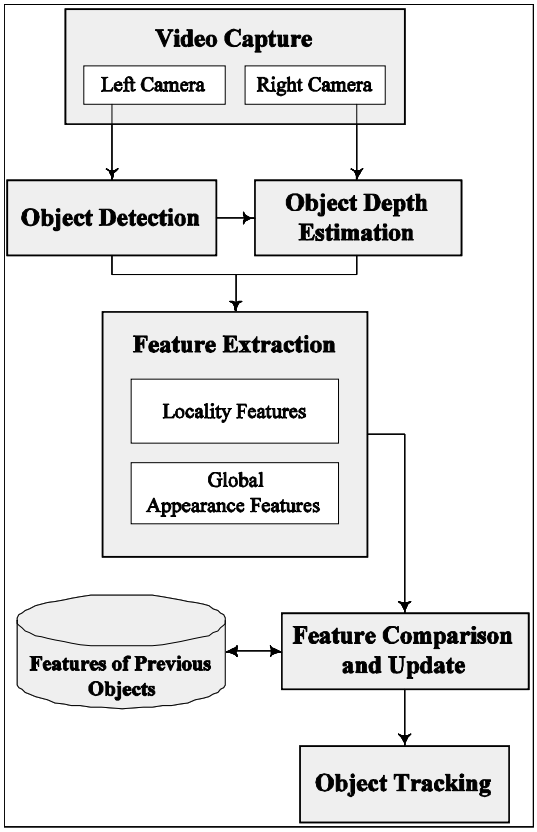


Figure 4.5.1.1. Diagram of the proposed framework.

4.5.1.3 Object Detection

Object detection plays a crucial role in many surveillance applications. Without robust detection of video objects, subsequent actions, such as object tracking and classification, would be infeasible. In video surveillance applications, segmentation should be able to overcome obstacles inferred by the presence of complex, moving background, which often occurs in the outdoor surveillance footage. In general, a video segmentation algorithm targeted for surveillance applications should:

- *Detect all objects of potential interest within a scene containing a complex, moving background*: A surveillance scene often contains a moving background such as wavering trees, moving clouds, flickering water surfaces, and similar. Thus, it is important that the algorithm can deal with such backgrounds and still detect the potential objects of interest.
- *Produce no false negatives and a minimal number of false positives*: A surveillance application generally prefers no false negatives, meaning that no potential threat is overlooked. On the other hand, having too many false positives makes potential post-processing activities, such as object classification, highly impractical.
- *Be fast and efficient, ensuring operation at a reasonable frame rate*: The object that poses a potential threat must be detected quickly so that the appropriate preventive action can be taken in a timely manner. Furthermore, if an algorithm operates at an extremely low frame rate due to its inefficiency, potential objects of interest could be overlooked.
- *Use a minimal number of scene-related assumptions*: When designing an object detection method targeted for surveillance applications, making the algorithm dependent upon too many assumptions regarding a scene setting (such as the location, presence, or absence of the scene objects like sky, land, forests, or buildings) may cause the algorithm to fail as soon as one or more of the assumptions do not hold.

Early methods for video object segmentation in the presence of a non-stationary background were based on smoothing the color of a background pixel over time using different filtering techniques, such as Kalman filters [3], to create a reference background frame. The reference frame was constantly updated and used to segment the foreground objects by subtracting it from the current frame of the input sequence. These methods are based on the assumption that movements of the background are much slower than those of the objects to be segmented and as such they are not effective for sequences with high-frequency background changes.
bnn/bnn Training and segmentation phases of BNNs used in our system for foreground-background video segmentation. RGB color values are used as features for classification.

More stable results were reported for methods utilizing a Gaussian-based statistical model whose parameters are recursively updated to follow gradual background changes within the video sequence [4]. More recently, this model was significantly improved by employing a *mixture of Gaussians* (MoG), in which the values of the pixels from background objects are described by multiple Gaussian distributions [5-7]. However, weak results are reported for video sequences containing non-periodical

background changes, often present in outdoor environments [8]. The problem with Gaussian-based models is that these models incorporate underlying assumptions about the probability density functions (PDFs) they are trying to estimate.

In 2003, Li *et al.* proposed a method for foreground object segmentation employing a Bayes decision framework where no a priori assumptions about the scene are necessary [9]. The method was proved effective even for the sequences containing complex variations and non-periodical movements in the background. First, a statistical model of for the changes between the current frame and the reference background image is established and maintained by applying an Infinite Impulse Response (IIR) filter to the sequence. Then, a Bayesian classifier is used to classify the changes detected through frame differencing between the current frame and the reference frame. This model does not impose any specific shape to the estimated PDFs.

Method by Li *et al.* uses binning of the features where a single probability is assigned to each bin, leading to a discrete representation of PDFs. This results in a large memory requirement so that only coarse resolutions, such as QCIF, are feasible. The approach of Li *et al.* has been adopted and extended to create a part of a surveillance system intended for maritime environments [10]. The results in this domain have been improved by altering the frame differencing step of the algorithm as well as using a color-based still image segmentation instead of the morphological operations in the post-processing of the background-subtraction results.

Since Bayesian models impose no constraints on the shape of the estimated probability density function, they are generally computationally expensive. However, using neural networks to implement Bayesian models, it is possible to make them suitable for parallel execution and increase their effectiveness.

The approach based on background modeling neural networks was proposed in [11]. The networks employ a biologically plausible implementation of Bayesian classifiers and nonparametric kernel-based density estimators. Results superior to those of Li *et al.* and MoG with 30 Gaussians were reported. The Background Modeling Neural Networks (BNN) address the problem of computational complexity of the kernel based background models by exploiting the parallelism of neural networks.

4.5.1.4 Object Detection with Background Modeling Neural Networks

A BNN [11] is a neural network designed to serve both as a statistical model of the background at each pixel position in the video sequences and as a highly-parallelized background-subtraction algorithm. The network is an unsupervised learner. It collects statistics related to the dynamic processes of pixel feature values changes. The learned statistics are used to classify a pixel either as foreground or background in each frame of the sequence.

Probabilistic motion (change) based background subtraction methods rely on the following supposition: pixel feature values corresponding to background objects will occur most of the time, i.e. more often than those pertinent to the foreground. Thus, if a classifier is able to effectively distinguish between the values occurring more frequently than others it should be able to achieve accurate segmentation. In a BNN

the segmentation problem is formulated to enable the use of Bayes decision rule to achieve segmentation as follows. For a certain frame t, one is trying to estimate the dependent variable $(\Theta_i \in \{f, b\})$. The event of pixel at location $i = (x_i, y_i)$ being part of the foreground corresponds to $\Theta_i = f$, while $\Theta_i = b$ when the pixel is pertinent to background. A Bayesian decision rule for simple pixel classification is formulated as:

$$\Theta = \begin{cases} f, & if \pi_{bi} P_{bi}(v) < \pi_{fi} P_{fi}(v); \\ b, & otherwise. \end{cases} \quad (1)$$

where $P_{bi}(v)$ is the PDF of background occurring at pixel i, $P_{fi}(v)$ is the PDF of background occurring at pixel i, π_{fi} and π_{bi} are prior probabilities of foreground and background occurring.

A central part of BNN is the classification subnet which contains four layers of neurons. *Input neurons* of this network simply map the inputs of the network, which are the values of the features for a specific pixel. Each input neuron is connected to all *pattern neurons*. The output of the pattern neurons is a nonlinear function of Euclidean distance between the input of the network and the stored pattern for that specific neuron. The only parameter of this subnet is the smoothing parameter of the Parzen estimator [12] used for estimating the PDFs P_{bi} and P_{fi}. The output of a single pattern neuron corresponds to the value of a single gaussian of the PDF estimation for the observed pixel value. The output of the summation units of the classification subnet is the sum of their inputs. The subnet has two *summation neurons*, each of them connected to all pattern neurons. The output values of the summation neurons correspond to initial Parzen estimates of P_{bi} and P_{fi} for the pixel value observed. These estimates are input to the last layer, containing a single *output neuron*. The final output of the network is a binary value indicating whether the pixel corresponds to foreground or background.

To form a complete background-subtraction solution a single instance of a BNN is used to model the features at each pixel of the image. The features used in our model are RGB color values, as depicted in Fig. 4.5.1.2.

In a parallel hardware-implementation, the speed of the BNN segmentation does not depend on the size of the frame. The delay of the network (segmentation time) corresponds to the time needed by the signal to propagate through the network and time required to update it. In a typical FPGA implementation this can be done in less than 20 clock cycles, which corresponds to a 2 ms delay through the network, for a FPGA core running at 100 ns clock rate. Hence, the BNN networks are capable of achieving a throughput of roughly 500 fps.

The core design of a BNN is provided in [11]. The basic BNN approach is recently improved in [13] by allowing an automatic selection of the width of kernels used to estimate the PDFs, thus making the entire BNN object segmentation process depend on a single control parameter.

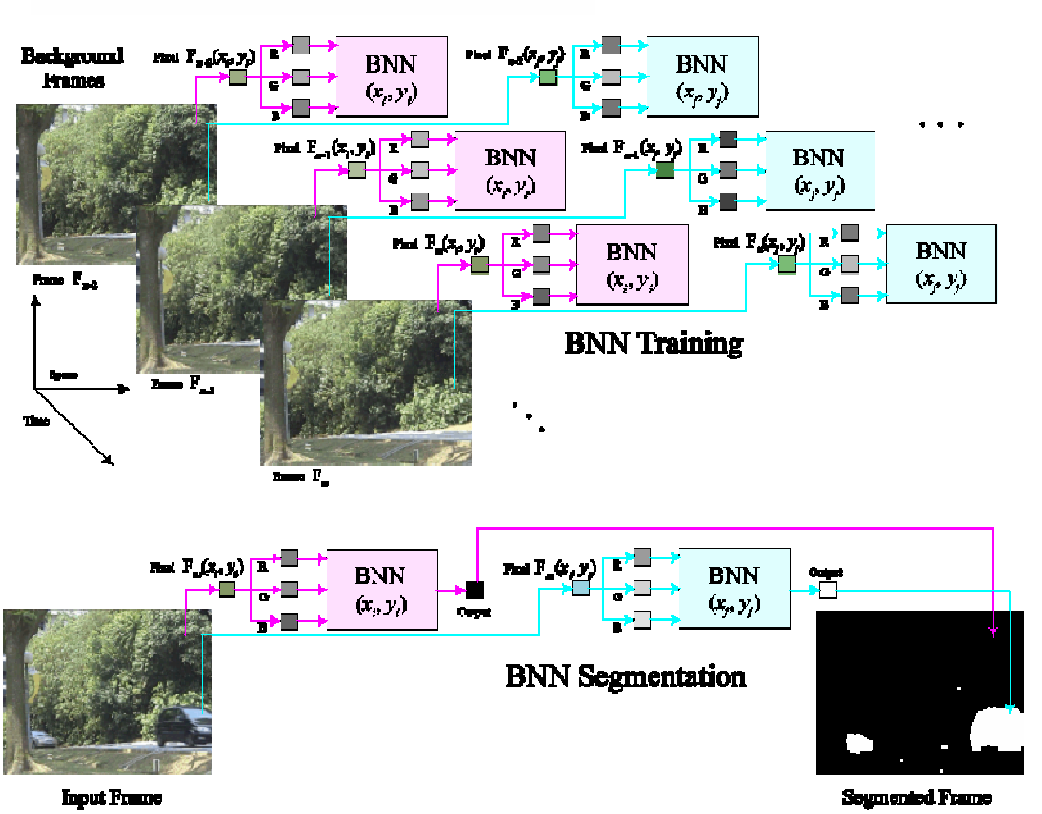


Figure 4.5.1.2. Training and segmentation phases of BNNs used in our system for foreground-background video segmentation. RGB color values are used as features for classification.

4.5.1.5 Removal of Shadows

Shadows associated with the moving foreground video objects are typically segmented along with the object. Unfortunately, the presence of shadows negatively affects the accuracy of feature extraction phase as it adds noise to the appearance features including shape, color and texture. Since these appearance features are used for object tracking in our system, a method for removing shadows from segmented objects is employed.

Shadow detection has been increasingly studied in the past years. According to the concluding remarks in a recent survey [14], there is no generally accepted method for removing shadows from a segmented video object. Instead, different approaches to shadow detection should be taken when addressing different kinds of applications. Several of the recently proposed approaches for shadow detections are designed to remove the shadows from the segmented video objects by removing the intensity values and exploiting the chromatic similarity between the shadow and the background image. The intensity values alone usually give no correlation information about the shadow and the background. Therefore, different color spaces are used, including YCbCr [15], YUV [16], HSV [17] and HSL [18].

In [18], a fast, real-time method for shadow removal is proposed where HSL (hue, saturation and lightness) color space is used to split the color information from the brightness values in the video object. When a video object is compared to the background image, all the pixel values that are similar in hue and saturation are considered pertinent to the shadow. This method assumes the following properties: (1) a shadow pixel is darker than the corresponding pixel in the background image, (2) the texture of the shadow is correlated with the corresponding texture of the background image. However, methods relying on these assumptions alone often increase the number of false negatives in the interior parts of an object, as illustrated in Fig. 4.5.1.3. Therefore, it is desired to limit the removal of potential shadow pixels only to the bordering parts of the objects.

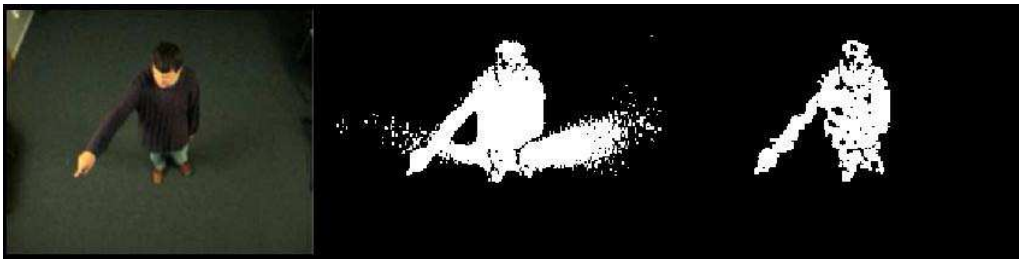


Figure 4.5.1.3. A sample result of the shadow removal algorithm from [18]: (a) the original frame, (b) the segmented video object with shadow, and (c) the segmented video object after shadow removal. Note that interior of the detected object is visibly affected by the algorithm.

Our modification to the HSL space-based shadow removal approach from [18] can be summarized as follows:

1. Convert color pixels of an object O to HSL color space
2. Obtain contour of O, denoted $C(O)$
3. Trace contour $C(O)$ and remove all pixels in O for which hue and saturation values are close to the corresponding hue and saturation values in the background image estimated in the BNN segmentation process according to the distance criteria from [18].
4. Repeat process from step 2 until no pixels are deleted from the contour.

The proposed contour-based approach is illustrated in Fig. 4.5.1.4. In each pass of the algorithm the shadow is gradually removed from the object without affecting its interior region. This method is employed in our detection and tracking system due to its effectiveness and fast performance. Fig. 4.5.1.5 shows a sample result of applying the proposed shadow removal method to a video object produced by the BNN segmentation.

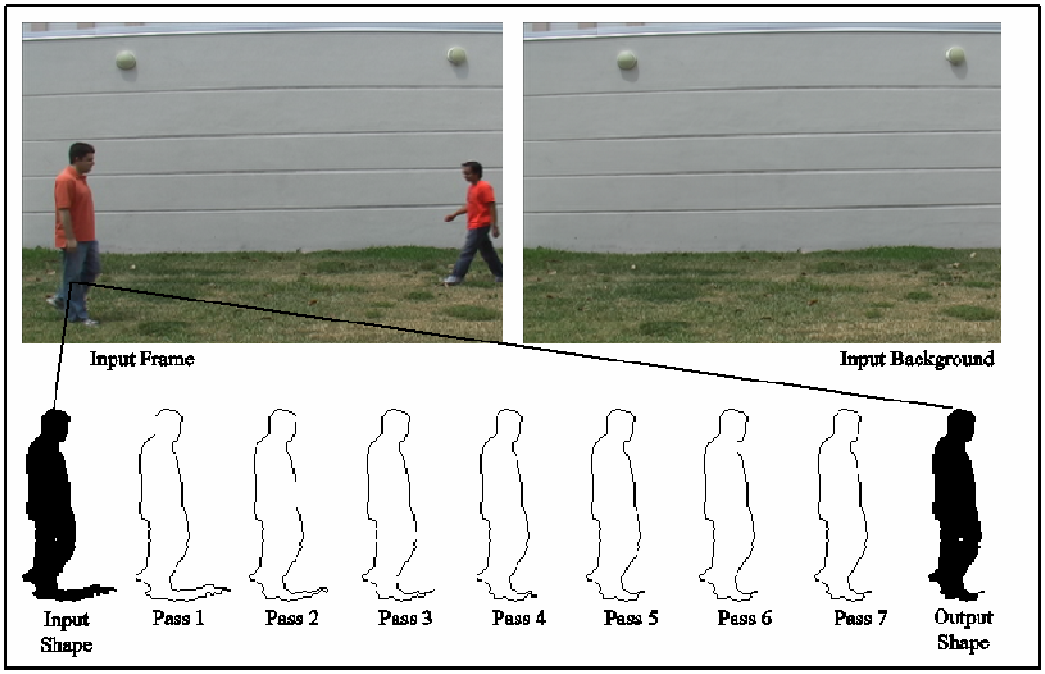


Figure 4.5.1.4. The proposed contour-based method broken down into phases to illustrate the gradual removal of shadows from the video object.

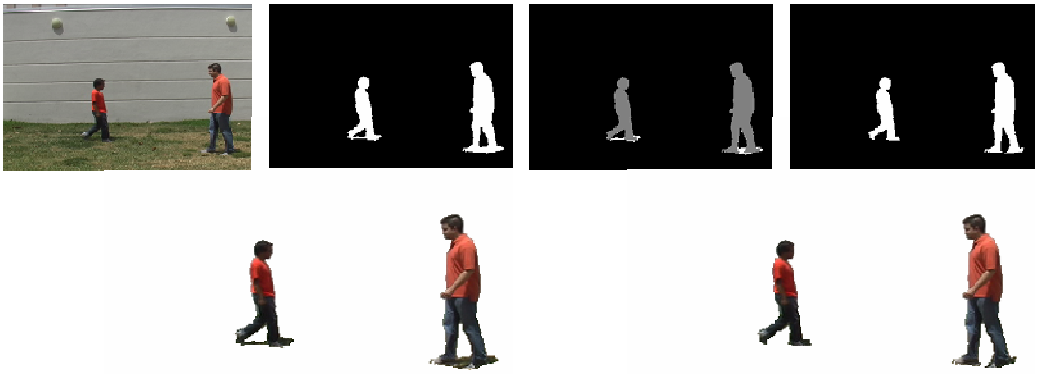


Figure 4.5.1.5. Segmentation results produced with BNN (a) without shadow removal; (b) with shadow detection and removal.

4.5.1.6 Object Depth Estimation from Stereo

Depth information regarding tracked video objects represents a powerful feature capable of distinguishing objects with similar color, shape and texture features. Depth can be estimated from a pair of stereo frames using a stereo correspondence algorithm.

Given a pair of stereo images, the correspondence problem refers to determining the *match sequence* for each corresponding scanline in a pair of stereo images. The match refers to an ordered pair (x, y), where x and y are the positions in same scanlines of

left and right stereo pair, respectively, such that the pixel values corresponding to these positions, $F^L(x)$ and $F^R(y)$, represent images of the same scene point. Here, it is assumed that the stereo images are properly aligned so that the scanlines are the epipolar lines. Unmatched pixels are labeled as being *occluded* and adjacent occluded pixels which are bounded by non-occluded pixels are referred to as an *occlusion*.
The disparity of a pixel position x in the left scanline that matches the pixel y in the right scanline is defined as the difference $x-y$, while the disparities of the pixels in an occlusion are assigned the farther of the two bounding regions. Approaches to the stereo correspondence problem construct the so called *disparity map*, which is also known as the *depth map* or the *depth estimation*, since it described the discrete estimation of the third spatial dimension.

Fig. 4.5.1.6 demonstrates the configuration we used to capture stereo images. Two cameras ``left" and ``right" were fixed and pointed at the same scene. The cameras were placed approximately ten centimeters apart on the same, level horizontal plane, creating a slight disparity between the image captured by the left camera and the corresponding image from the right camera. This disparity was then used to estimate the depth of objects in the captured scene.

Unfortunately, there is a tradeoff between speed and quality of depth reconstruction among different algorithms for a general depth estimation from a stereo pair of images. The most recent algorithms that reportedly achieve the closest depth estimation to the ground truth depth are based on computationally complex procedures involving over-segmentation of color images [19] and [20]. For example, an approach from [19] reportedly takes between 14 and 25 seconds on 2.1 GHz general-purpose 64-bit processor to estimate depth from a pair of CIF stereo images. Ranking of many algorithms in terms of accuracy is available in [21] as well as at the constantly updated Middlebury stereo web page http://vision.middlebury.edu/stereo.

Faster algorithms rely on less accurate estimation. In [22], Birchfield and Tomasi proposed an approach to estimating disparity map from stereo images using comparisons of pixel values in corresponding scanlines in left and right images. A typical PC implementation of Birchfield-Tomasi algorithm takes about few seconds for a stereo pair of CIF images with smaller selection of maximum disparity search parameter d_{max} (e.g. 20 or less). The approach is considered fast and it is based on pixel-to-pixel matching in a scanline.

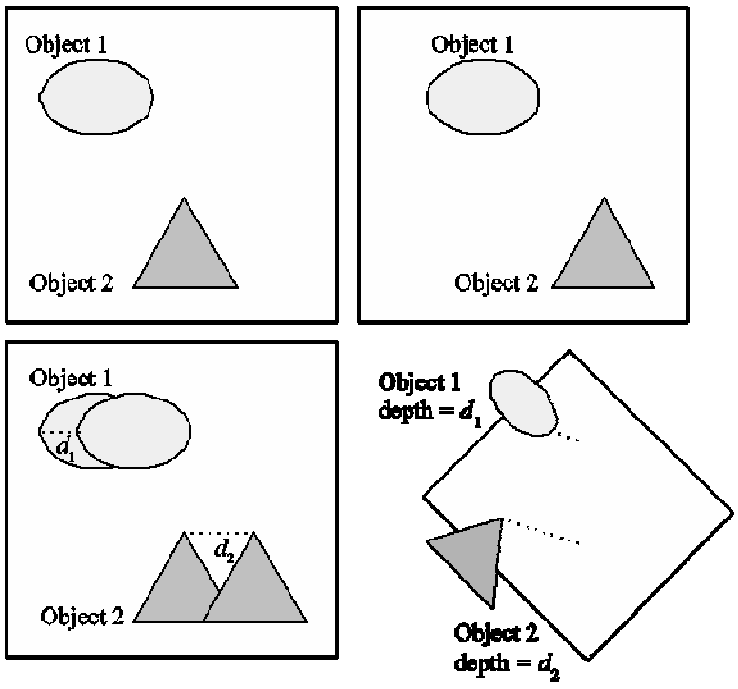


Figure 4.5.1.6. Process of estimating depth of the entire video object using disparity measure from left and right stereo frames (top two figures).

However, the general problem of stereo correspondence brings an unnecessary computational overload when our system is concerned, since we do not need to perform stereo matching on the background. Instead, we restrict to a problem of finding a depth of a segmented object as a whole, which is computationally much simpler problem and could be solved in a straightforward region shift matching that is more than capable of achieving depth estimation in a real-time. In our system, we use a disparity computation based on this principle, as depicted in Fig. 4.5.1.6.

Let F^L and F^R denote the n^{th} frame in the synchronized sequences captured with left and right camera, respectively. The disparity d of an object $O \in F^L$ produced by a BNN segmentation of the left frame is determined as follows:

$$d(O) = \min_{0 \le i \le d_{max}} \left(\sum_{j=0}^{bh} \sum_{k=0}^{bw} s_{i,j,k} \right),$$

$$s_{i,j,k} = \begin{cases} |B(O)_{k,j} - F^R_{bx+k-i,by+j}|, & if S(O)_{k,j} = 1; \\ 0, & otherwise, \end{cases}$$

where $|\cdot|$ denotes the absolute value, $S(O)$ denotes the binary shape of O, $B(O)$ the bounding box of O, while bw, bh and (bx, by) denote their width, height, and the coordinates of their top-left corner within F^L, respectively.

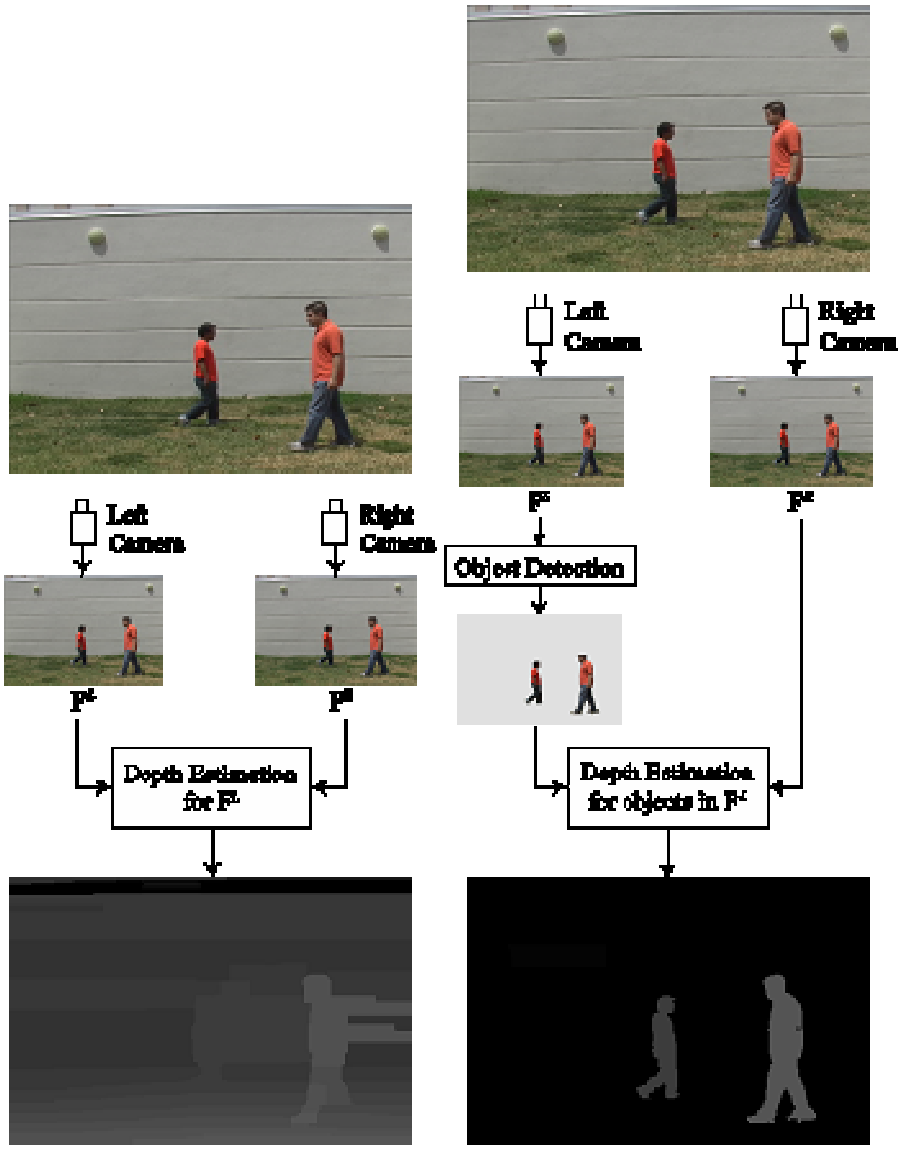


Figure 4.5.1.7. Depth estimation with: (a) general stereo correspondence method (in this instance Birchfield-Tomasi algorithm from [22]), and (b) the proposed object-based stereo correspondence method. The running time of (a) is in seconds while the running-time of (b) is in milliseconds on a standard PC.

This process allows us to use depth as one of the strong discriminatory features needed for successful tracking of detected objects. Fig. 4.5.1.7 shows the steps involved in a general-purpose depth estimation in which the disparity map for the entire left frame is estimated and an object-oriented disparity estimation employed in the proposed system.

4.5.1.7 Feature Extraction

In order to successfully track video objects across multiple frames, a set of representative features must be extracted and compared with previously encountered objects. There are two types of features extracted in the proposed system: (1) *locality features*, such as depth, size and position, and (2) *global appearance features*, such as color, shape and texture. The locality features include all features of an object that are dependant on object's local properties which could potentially change during tracking. These features are likely variant to 3D isometric transformations including rotation, translation and scaling. On the other hand, global appearance features are likely not to change during tracking, and are robust to isometric transformations (to a certain degree). While shape is arguably not invariant over tracking time for non-rigid objects, it is likely periodic so that its ``invariance'' is captured at certain time intervals.

Due to the nature of surveillance applications, the extracted features are desired to have small computational cost, small storage requirements, and, in case of appearance features, be invariant to translation, rotation and scaling. Since most of the MPEG-7 descriptors corresponding to object's global appearance follow these requirements, a number of features used in our system are either entirely or partially based on these standard descriptors. While similarity metrics for descriptors are not defined in the MPEG-7 standard, we adopted several commonly used distance metrics which are proven effective in our experiments. In addition to descriptors provided by MPEG-7 and depth feature provided by the depth estimation algorithm, we use a few additional easily computable locality features.

4.5.1.8 Locality Features

In our system, the size of a detected object is characterized by its bounding box size. The bounding box size is represented as a two dimensional vector (Φ_{bw}, Φ_{bh}) where Φ_{bw} denotes the width and Φ_{bh} the height of object's bounding box. The position of object O is represented by the *centroid* of its polygonal approximation with n equidistant points, $(x_i, y_i), i = 0, \ldots, n-1$, randomly selected from its contour $C(O)$. The centroid, also known as the *center of gravity* or the *center of mass*, is given as a point (Φ_{cx}, Φ_{cy}) as follows:

$$\Phi_{cx} = \frac{1}{6A} \sum_{i=0}^{n-1} (x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i),$$

$$\Phi_{cy} = \frac{1}{6A} \sum_{i=0}^{n-1} (y_i + y_{i+1})(x_i y_{i+1} - x_{i+1} y_i),$$

$$A = \frac{1}{2} \sum_{i=0}^{n-1} (x_i y_{i+1} - x_{i+1} y_i),$$

where $(x_n, y_n) = (x_0, y_0)$ to create an imaginary closed polygon with A representing its area. We set n to 10 in our experiments therefore working with the centroids of decagonal approximations of objects' shapes.

Our depth estimation algorithm produces an estimation of depth for each object in a frame. Object depth, denoted by Φ_d, is represented as the disparity value calculated for an entire object. The value of d_{max} was set to 31, thus allowing 32 depth planes and limiting the depth feature size to 5 bits.

The distance between two locality features is measured using the standard Euclidean distance $\|\cdot\|$. Thus the following metrics are used to measure the distance of locality features of objects O_1 and O_2:

$$D_i = \| \Phi_i(O_1) - \Phi_i(O_2) \| = | \Phi_i(O_1) - \Phi_i(O_2) |,$$

where $i \in \{bw, bh, cx, cy, d\}$.

4.5.1.9 Global Appearance Features

Appearance features are characterized by MPEG7-like color, shape and texture descriptors.

4.5.1.9.1 Color Features

In our system we used a feature based on MPEG-7 dominant color descriptor for object tracking. This descriptor specifies a set of dominant colors in an arbitrarily shaped region (object) [23].

The following subset of feature components is selected from the actual MPEG7 dominant color descriptor: number of dominant colors (up to 8), indices of dominant colors in RGB color space quantized to 3 bits per channel, and percentages of dominant colors.

The distance metric D_c we use for our color descriptor is based on a *quadratic histogram distance measure* (QHDM) from [24]. If color feature $\Phi_c(O_j)$ of a video object O_j is denoted by its size n_j, three dimensional indices of dominant colors c_{ji} and corresponding percentages p_{ji}, then the distance metric between two features $\Phi_c(O_1)$ and $\Phi_c(O_2)$ is defined as:

$$D_c = \sum_{i=1}^{n_1} p_{1i}^2 + \sum_{j=1}^{n_2} p_{2j}^2 - \sum_{i=1}^{n_1}\sum_{j=1}^{n_2} 2a_{ij} p_{1i} p_{2j},$$

where a_{ij} is the similarity coefficient between colors c_{1i} and c_{2j}. The similarity coefficient is defined as:

$$a_{ij} = \begin{cases} 1 - \| c_{1i} - c_{2j} \| / \alpha T_d, & \| c_{1i} - c_{2j} \| \le T_d; \\ 0, & \| c_{1i} - c_{2j} \| > T_d, \end{cases}$$

where $\|\cdot\|$ denotes standard Euclidean norm and T_d the threshold for maximum distance allowed for similar colors, with control rate parameter α. For our dominant color descriptor we used 0.5 and 0.05 for T_d and α, respectively.

4.5.1.9.2 Shape Features

The MPEG-7 standard provides three shape descriptors: region shape descriptor, contour shape descriptor and 3-D shape descriptor. 3-D shape descriptor is not used in our system since we deal with 2-D projections of the real-world captured by fixed stereo cameras where 3-D shapes are unknown. Our system utilizes both region and contour shape descriptors since they provide mutually exclusive concise discriminatory shape information about the object.

The region shape descriptor uses a complex 2-D Angular Radial Transform (ART) coefficients defined on a unit disk in polar coordinates as:

$$F_{nm} = \int_0^{2\pi}\int_0^1 V_{nm}(\rho,\theta), f(\rho,\theta)\rho\, d\rho d\theta,$$

where $f(\rho,\theta)$ is an image function in polar coordinates, and $V_{nm}(\rho,\theta)$ is the ART basis function separable along the angular and radial directions:

$$V_{nm}(\rho,\theta) = A_m(\theta)R_n(\rho),$$

$$A_m(\theta) = \frac{1}{2\pi}\exp(jm\theta),$$

$$R_n(\rho) = \begin{cases} 1, & n = 0; \\ 2\cos(\pi n\rho), & n \neq 0. \end{cases}$$

According to the MPEG-7 standard [23], ART coefficients are calculated for 3 radial and 12 angular directions ($0 \le n < 3$, $0 \le m < 12$). The coefficients are first normalized for scale invariance by dividing each coefficient with V_{00}, and then quantized to a 4-bit representation, denoted by *MagnitudeOfART*, using a non-uniform binning.

A total of 35 4-bit *magnitude of ART* values are stored as features, thus requiring only 17.5 bytes of storage. The descriptor captures well characteristic features of the shape, enabling similarity-based retrieval. It is robust to partial occlusion of the shape and to perspective transformations (rotation, translation and scaling). In our system, it is used to characterize the object's shape and to determine the periodicity in its motion.

We measured the distance between two region shape descriptors using the Euclidean distance as

$$D_s = \sqrt{\sum_{i=1}^{35} (\Phi_s(O_1)[i] - \Phi_s(O_2)[i])^2},$$

where $\Phi_s(O_j)[i]$ is the i^{th} magnitude of ART value of region shape descriptor of O_j.

Contour shape MPEG-7 descriptor describes a closed contour of an object or region. It is based on the Curvature Scale Space (CSS) representation of the contour.

A number of equidistant points are arbitrarily selected on the contour, and two series created: X comprising of grouped x-coordinate values, and Y containing all y-coordinate values. The contour is then gradually smoothed by a repetitive application of a low-pass filter with the kernel $(0.25,0.5,0.25)$ to X and Y. The filtering process is terminated when the contour becomes convex since the concave parts eventually flatten-out.

Vertical coordinates, denoted by y_{css}, are defined as the number of passes of the filter in the given point. Contour curvature function zero-crossing points separate concave and convex parts of the contour. Each zero-crossing is marked on the horizontal line corresponding to the smoothed contour and at the x_{css} location corresponding to the position of this zero-crossing along the contour. The CSS image has characteristic peaks. The coordinate values of the prominent peaks (x_{css}, y_{css}) in the CSS image are extracted. The peaks are ordered based on decreasing values of y_{css}, transformed using a non-linear transformation and quantized. Finally, the eccentricity and circularity of the contour are also calculated, quantized and stored. The circularity is defined as:

$$circularity = \frac{perimeter^2}{area},$$

while eccentricity is given by

$$eccentricity = \sqrt{\frac{i_{20} + i_{02} + \sqrt{i_{20}^2 + i_{02}^2 - 2i_{20}i_{02} + 4i_{11}^2}}{i_{20} + i_{02} - \sqrt{i_{20}^2 + i_{02}^2 - 2i_{20}i_{02} + 4i_{11}^2}}},$$

$$i_{02} = \sum_{k=1}^{N} (y_k - y_c)^2,$$

$$i_{11} = \sum_{k=1}^{N} (y_k - y_c)(y_k - y_c),$$

$$i_{20} = \sum_{k=1}^{N} (x_k - x_c)^2,$$

where N is the number of points inside the contour shape, and (x_c, y_c) is the center of mass of the shape.

In our system, we store and use only global curvature which describes circularity Φ_{gc} and eccentricity Φ_{ge} of the object's contour $C(O)$. Euclidean distances (which are in this instance equivalent to the absolute differences), denoted D_{gc} and D_{ec}, are used to measure the distances between global curvature features of two video objects.

4.5.1.9.3 Texture Features

In our system MPEG-7 edge histogram descriptor is used to characterize object's texture. This descriptor, denoted as Φ_t, specifies the spatial distribution of five types of edges in local image regions. There are four directional edges (horizontal, vertical, 45 degree, and 135 degree edge) and one non-directional edge in each local region called a sub-image. For each sub-image a local edge histogram with 5 bins is generated. Since there are five types of edges for each sub-image, we have a total of $16 \times 5 = 80$ histogram bins.

Thus, the descriptor consists of 80 3-bit values called *bin count*. These values are non-linearly quantized to a 3-bit value corresponding to counts of 5 different types of edges within each of the 16 sub-images.

The distance between texture features of objects O_1 and O_2 is defined as the Euclidean distance between the bin counts:

$$D_t = \sqrt{\sum_{i=1}^{80} (\Phi_t(O_1)[i] - \Phi_t(O_2)[i])^2}, \tag{2}$$

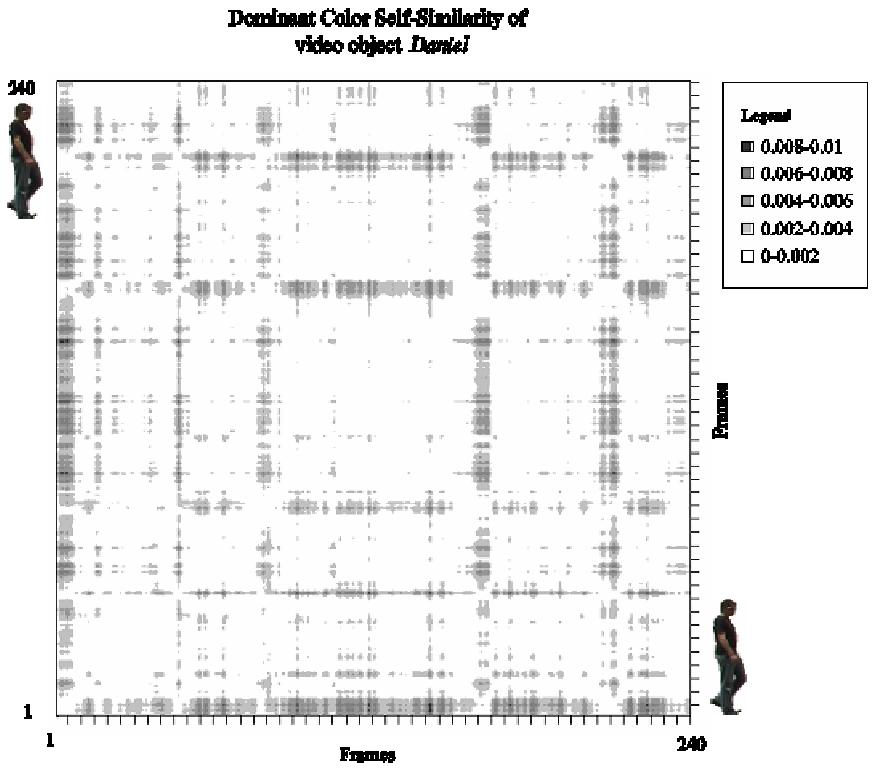
where $\Phi_t(O_j)[i]$ is the value of the i^{th} edge histogram bin count of the texture feature of object O_j.

4.5.1.10 Tracking of Video Objects based on Feature Update and Comparison

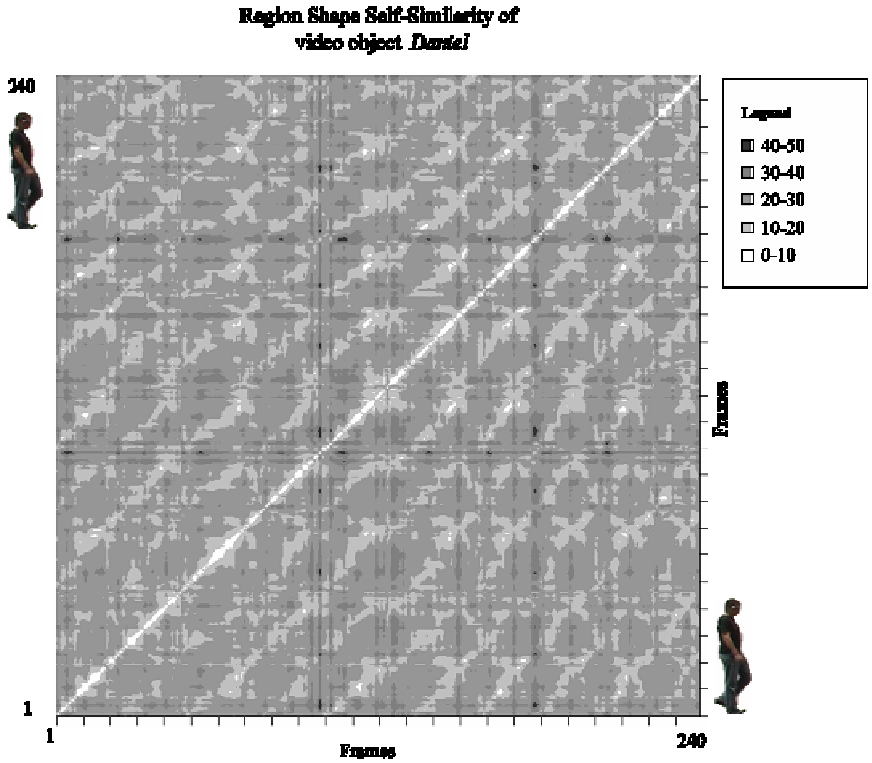
Tracking of detected video objects across multiple frames refers to distinguishing all instances of a given object in the successive frames from the point the object enters the observed scene until it leaves it. Tracking *difficulties* could arise when the tracked object occludes with other objects during tracking, gets obstructed by the background, or when scene is cluttered with objects that have similar features. When no tracking difficulties are present, tracking of objects essentially becomes a correspondence problem between objects in the previous frame and objects in the current frame.

To solve the frame-to-frame object correspondence, we rely on the assumption that both locality and global appearance features of the corresponding objects are similar from frame to frame since the capturing of frames typically occurs at very high frequencies (such as 25 or 30 Hz) thus allowing only relatively small changes in size, position, depth and overall appearance of an object. Fig. 4.5.1.8 (a, b and c) shows the

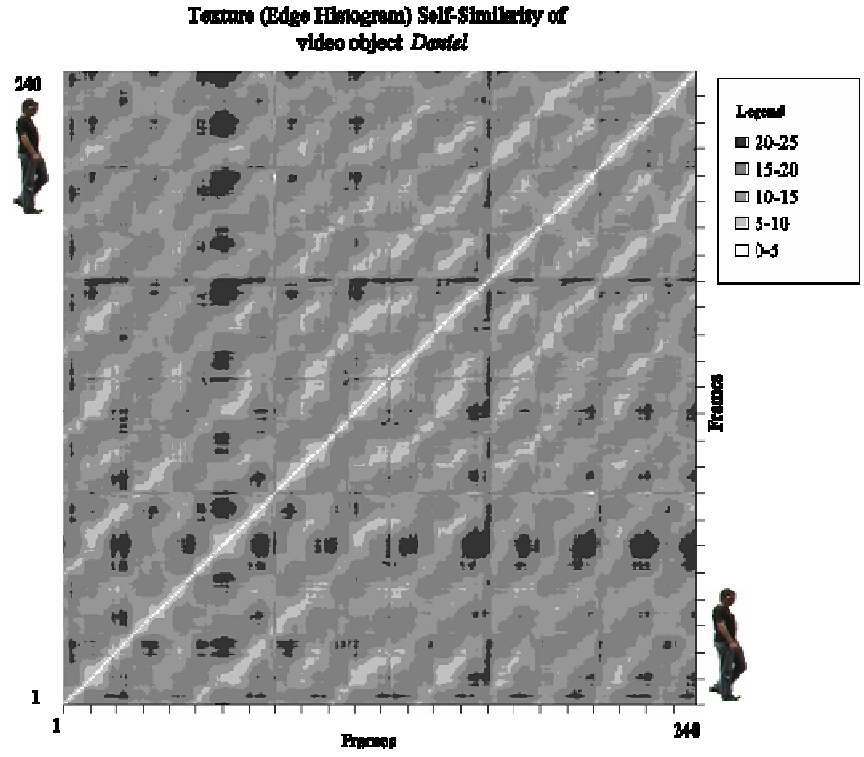
self-similarities of various features of a video object during tracking. It can be observed that smallest distances occur around the diagonal of the matrix indicating that the largest similarities occur in frames close to one another. A more complete comparison of overall and frame-to-frame self-similarities of four different video objects from the experimental suit of stereo sequences is summarized in Table 4.5.1.1, where $\overline{D_\phi}$ and $\overline{D_\phi^\varphi}$ denote the average overall and frame-to-frame distances between feature ϕ of the same tracked object.



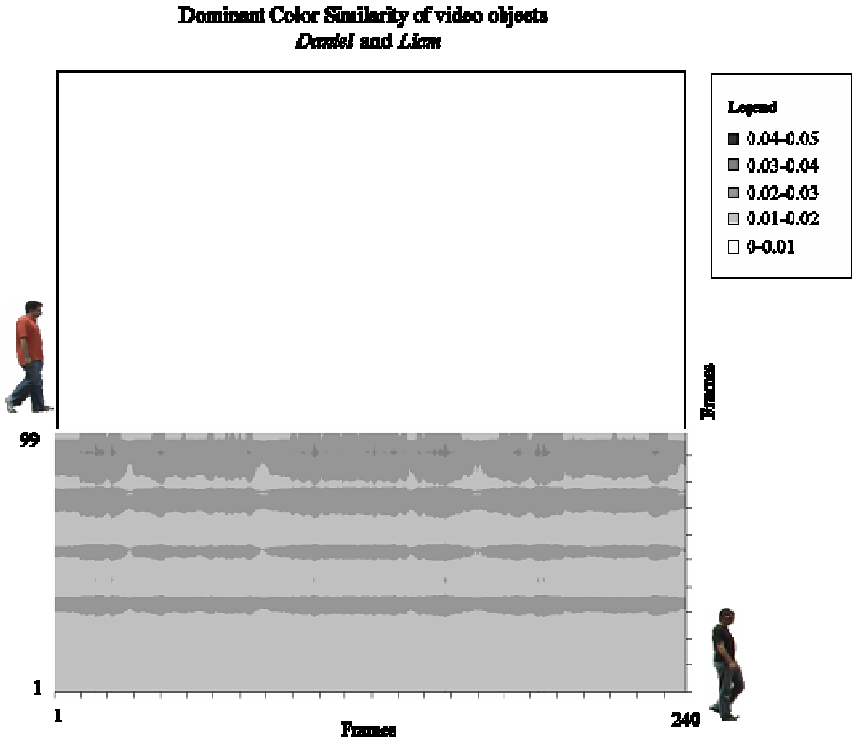
(a)


Region Shape Self-Similarity of
video object Daniel
240
1
1
240
Frames
Frames
Legend
40-50
30-40
20-30
10-20
0-10

(b)


Texture (Edge Histogram) Self-Similarity of
video object Daniel
240
1
1
240
Frames
Frames
Legend
20-25
15-20
10-15
5-10
0-5

(c)



(d)

Figure 4.5.1.8. Feature distances of video object during tracking: (a), (b) and (c) show self-similarity of color, region shape and texture features for object *Daniel*, respectively, while (d) shows the similarity of color features between object *Daniel* and object *Liam.*

Table 4.5.1.1. Comparison of objects' overall and frame-to-frame average self-similarity (distances) for several features.

				
	Daniel	*Lakis*	*Liam*	*Carlos*
$\overline{D_c}$	0.001	0.001	0.009	0.006
$\overline{D_c^{\varphi}}$	0.00032	0.00012	0.00046	0.00043
$\overline{D_s}$	23.108	23.743	24.694	24.400
$\overline{D_s^{\varphi}}$	13.684	14.213	13.866	16.000
$\overline{D_{gc}}$	7.789	10.177	7.116	7.961
$\overline{D_{gc}^{\varphi}}$	2.715	3.137	2.255	2.640

$\overline{\mathsf{D}_{gc}}$	5.391	5.446	5.127	6.015
$\overline{\mathsf{D}^{\varphi}_{ge}}$	1.159	1.362	1.193	1.520

In order to measure the frame-to-frame self-similarity of a video object O, a self-similarity vector $\langle\overline{\mathsf{D}^{\varphi}_{\phi}}(\mathsf{O})\rangle$, $\phi\in\{bw,bh,cx,cy,d,c,s,gc,ge,t\}$, is constructed and constantly updated using the extracted features in those frames in which O is not occluded with other tracked objects in the scene (and more generally, where there are no tracking difficulties observed for O).

Without loss of generality, we assume that initially no objects are contained within the surveillance scene. When an object first enters the scene, it is labelled as new. A new object given a unique object ID and its features are extracted and stored. Also, a new self-similarity vector $\langle\overline{\mathsf{D}^{\varphi}_{\phi}}(\mathsf{O})\rangle$ is formed with its values set to 0. In the next frame F_2, the tracking system must establish the correspondence between the newly detected object O^1 in the previous frame F_1 and its corresponding image O^2 in the current frame. The following correspondence method is only used for matching the second instance of the newly detected object. If $\mathsf{O}^2_1,\ldots,\mathsf{O}^2_n$ is the list of objects in the current frame, then O^1 is identified as O^2_j for that $j\in\{1,\ldots,n\}$ for which the distance of their features is minimal according to the ``majority wins" principle. The ``majority wins" principle is used since the feature distance measures are not normalized. In ``majority wins" principle the vectors of feature distances are compared among themselves coordinate-wise (feature-wise) so that each feature is separately compared with same feature in other vectors and a sorting index given in ascending order. The vector with smallest sum wins as the closest one.

For each tracked object O its self-similarity vector $\langle\overline{\mathsf{D}^{\varphi}_{\phi}}(\mathsf{O})\rangle$ is updated from its $(n-1)^{\text{th}}$ instance to n^{th} using the following recurrence relation defined in [25]:

$$\overline{\mathsf{D}^{\varphi}_{\phi}}(\mathsf{O})^{(n)}=\overline{\mathsf{D}^{\varphi}_{\phi}}(\mathsf{O})^{(n-1)}+\frac{\mathsf{D}_{\phi}(\mathsf{O},\mathsf{O})-\overline{\mathsf{D}^{\varphi}_{\phi}}(\mathsf{O})^{(n-1)}}{n}.$$

The method based on ``majority wins" principle, as well as the general method of measuring the Mahalanobis distance to accommodate for differently skewed feature values [26], do not fully utilize the discriminatory information inherited in objects' self-similarity. Thus, a similarity measure, denoted as S, is proposed for solving the object's frame-to-frame correspondence problem *after* the second instance of the tracked object. When no tracking difficulties are detected, the object O^N_j from N^{th} frame is matched with an object from previous frame O^{N-1}_i as follows:

$$\mathsf{O}^N_j:\mathsf{O}^{N-1}_i,\ \text{if}\ \ \mathsf{S}(\mathsf{O}^{N-1}_i,\mathsf{O}^N_j)\geq T_{\mathsf{S}},$$

$$\mathsf{S}(\mathsf{O}_1,\mathsf{O}_2)=\sum_{i=1}^{k}\lambda_i(\mathsf{O}_1,\mathsf{O}_2),$$

$$\lambda_i(\mathsf{O}_1,\mathsf{O}_2)=\begin{cases} w_i, & \mathsf{D}_{\phi_i}(\mathsf{O}_1,\mathsf{O}_2)\le\rho_i; \\ 0, & \textit{otherwise.} \end{cases}$$

where $\langle\phi_1,\ldots,\phi_k\rangle$ are the used features, w_i is the weight associated with feature ϕ_i, and ρ_i is the close proximity radius for features ϕ_i. In our implementation, $\langle\phi_1,\ldots,\phi_k\rangle=\langle bw,bh,cx,cy,d,c,s,gc,ge,t\rangle$ so $k=10$, w_i's are all set to $1/k$, and close proximity radius is set to be linearly dependant on the current average frame-to-frame feature distance $\rho_i=a\overline{\mathsf{D}_{\phi_i}^{\varphi}}+b$, where $a=2$ and $b=1$, except for the color feature where $b=0.001$ since it uses a non-Euclidean distance. The final parameter T_S controls the lowest percentage of observed similarity needed for a match. We used the value of 0.7 for T_S in our simulation meaning that objects are matched if at least 70% of their features are close.

When there are no matches for object O_i^{N-1}, the system checks for possible occlusion. The occlusion is checked by observing the projected trajectory of O_i^{N-1} using the shift of O_i^{N-1} from frame $N-2$ to $N-1$. The shift is given by its position features as $(\Phi_{cx}(\mathsf{O}_i^{N-1})-\Phi_{cx}(\mathsf{O}_i^{N-2}),\Phi_{cy}(\mathsf{O}_i^{N-2})-\Phi_{cy}(\mathsf{O}_i^{N-1}))$. If the bounding box of the shifted O_i^{N-1} overlaps with the bounding box of any of the objects in frame N, then the occlusion is detected and objects pertinent to the occlusion are labelled as occluded. The detected occlusions in our system are tracked just like any other objects except with an additional information as to which previously tracked objects constitute the occlusion blob.

On the other hand, if there is no match for object O_j^{N} in the system, it is concluded that either O_j^{N} *split* from the occlusion or that it is a *new* object. If O_j^{N} is in the close proximity of any of the currently tracked occlusions, occlusion splitting is discerned. Otherwise the object is tracked as new.

When occlusion splitting occurs, one must be able to determine which of the object contained in the occlusion split from the occlusion blob K. To accomplish this, given the knowledge and features of objects constituting the occlusion, a subset of features denoted by *stable features* is used. Stable features are such features of a tracked object that are either roughly constant or periodic in nature. Global appearance features are considered stable due to their invariance to isometric transformations. As far as locality features are concerned, while position is not considered stable, depth and size features are considered stable only if their rough constancy is observed, which is the case for objects moving approximately perpendicular to the camera view. The concept of stable features follows the assumption that frame gap between observed features of an object formed as a result of object's occlusion is too large to use all features for

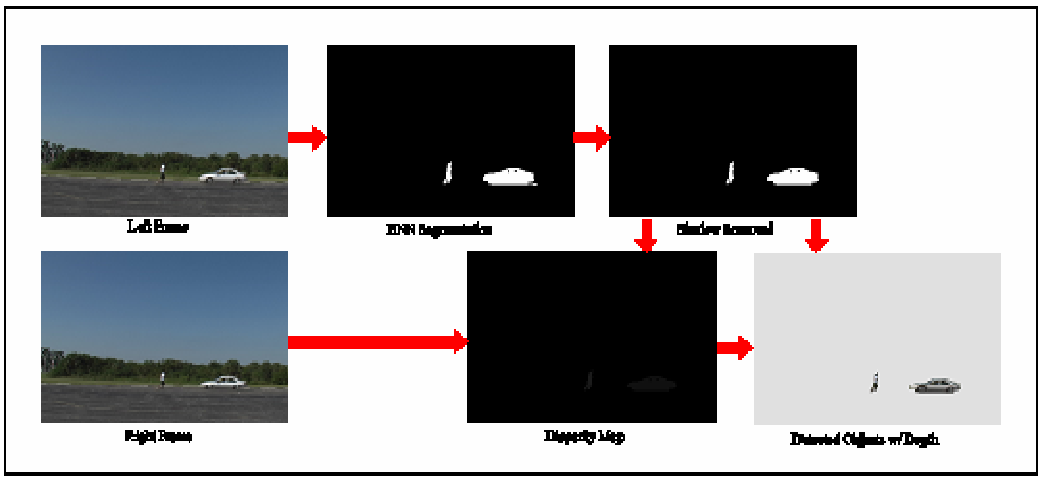
similarity measure as it is the case for the frame-to-frame similarity, as illustrated in Fig. 4.5.1.8 and Table 4.5.1.1.

In our system, the following is used to determine which of the objects from K is in correspondence with O_j^N. First, the *stable* features of O_j^N are compared to *stable* features of all tracking instances of O_i and using the ``majority wins" principle the instance of O_1 with the smallest distance is selected, denoted d_1. This follows for all other objects from K, ending with a set of distances $d_1 \ldots d_n$. Finally if $d_i = \min(d_1, \ldots, d_n)$, then O_j^N is set to correspond to object O_i. The self-similarity vector for O_i is at that point updated accordingly with feature information from O_j^N.

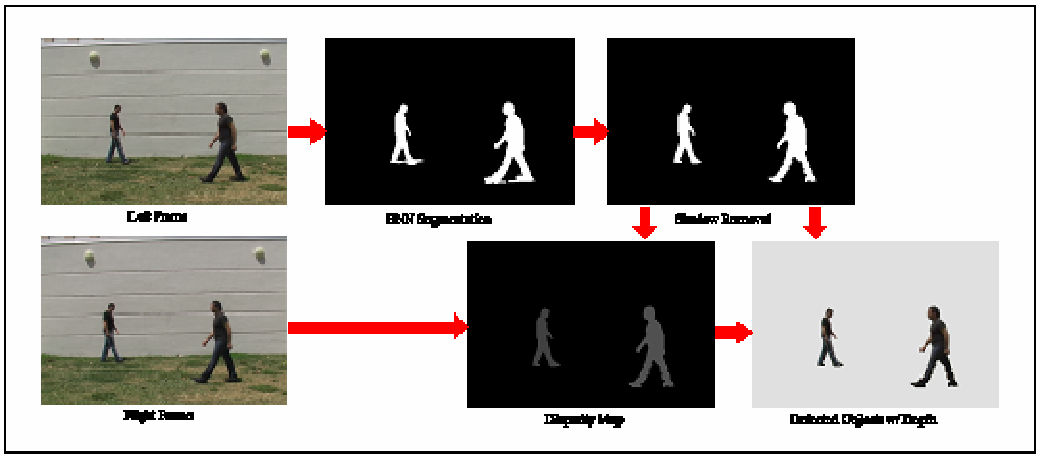
4.5.1.11 Experiments and Results

To evaluate the performance of the proposed object detection and tracking a PC based application has been developed. Previously, experiments have been conducted on a set of sequences with complex backgrounds in order to evaluate the neural network-based segmentation [11]. Since our method requires stereo sequences, and no such sequences are available as a benchmark, we provide experimental results on a set of our surveillance-like stereo sequences taken outdoors.

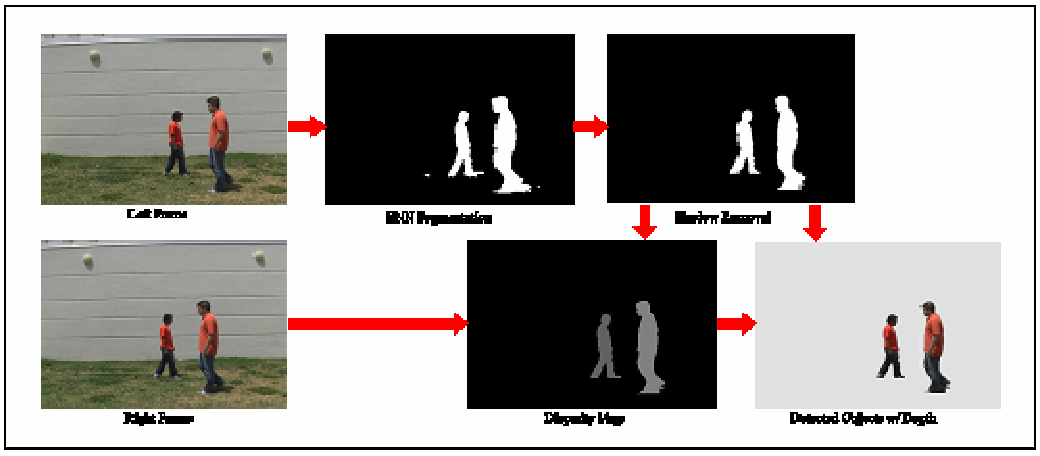
The combined object detection results for five surveillance-like outdoor sequences are illustrated in Fig. 4.5.1.9. For each of the five sequences we show the original frame, its segmentation result before and after the shadow removal, and detected objects along with their estimated depth. Accurate segmentation using a combination of BNN approach and an effective shadow removal can be observed in Fig. 4.5.1.9. In addition, objects' depth also appear to be estimated accurately.
Fig. 4.5.1.10 shows the tracking results for our set of experimental sequences. In our system, each newly detected object is assigned different color. Also, we ran experiments on sequences in the order presented in Fig. 4.5.1.10 and we kept the features of detected objects from previously run sequences. It can be observed from Fig. 4.5.1.10 that all different objects have been correctly identified for tracking (since their tracking colors are different). Also, for each currently tracked object, a trajectory is shown in corresponding color. Robustness of tracking in the presence of object occlusion can be observed even in cases when the tracked objects have similar color, shape and texture to a human eye (in this case two walking persons in the same outfit).



(a) Object detection in the *Alvaro and Pale Car* sequence

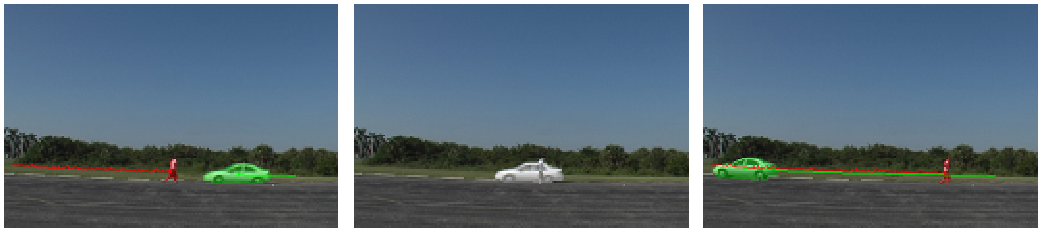


(b) Object detection in the *Daniel and Lakis* sequence



(c) Object detection in the *Carlos and Liam* sequence

Figure 4.5.1.9. Detection of video objects and their depth in several experimental outdoor sequences sequences. Red arrows are used to indicate the sequential dependency of different detection phases.



(a) Tracking results for the *Alvaro and Pale Car* sequence before, during and after object occlusion

(b) Tracking results for the *Pale Car and Dark Car* sequence before, during and after object occlusion

(c) Tracking results for the *Daniel and Lakis* sequence before, during and after object occlusion

(d) Tracking results for the *Liam and Carlos* sequence before, during and after object occlusion

(e) Tracking results for the *Carlos and Liam* sequence before, during and after object occlusion

Figure 4.5.1.10. Tracking of detected video objects based on their locality features (depth, size and position) and global appearance features (color, shape and texture) in several different outdoor sequences. The results demonstrate robustness in the presence of outdoor conditions and stable tracking after object occlusion.

4.5.1.12 Concluding Remarks

We developed a system for robust detection and tracking of video objects in stereo suitable for intelligent video surveillance applications. The method utilizes video segmentation based on Bayesian neural networks to achieve accurate and fast segmentation of video objects. The segmented results are further improved by proposed shadow removal algorithm capable of accurately removing shadows from detected video objects in real-time, thus improving the accuracy of objects appearance features. We also proposed a stereo correspondence algorithm that is capable of estimating depth on the object level in real-time from a single segmentation preprocessing. Finally we defined a set of localized and global appearance features that are used to solve the tracking correspondence. The features are compared in an adaptive manner and capable of maximizing the discriminatory information among tracked objects. The method is also able to detect occlusion as well as to resolve object's splitting from occlusion and continue tracking it. A number of experimental results are generated and results demonstrate robust tracking and detection of video objects from several stereo sequences.

References for Section 4.5.1

[1] R. Collins, A. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, and O. Hasegawa, "A system for video surveillance and monitoring," in *CMU-RI-TR*, 2000.
[2] I. Haritaoglu, D. Harwood, and L. Davis, "W4: Real-time surveillance of people and their activities," *PAMI*, vol. 22, no. 8, pp. 809–830, August 2000.
[3] K. P. Karmann and A. von Brandt, "Moving object recognition using an adaptive background memory," in *Timevarying Image Processing and Moving Object Recognition, 2, pp. 297-307*. Elsevier Publishers B.V.,1990.
[4] T. Boult, R. Micheals, X.Gao, P. Lewis, C. Power, W. Yin, and A. Erkan, "Frame-rate omnidirectional surveillance and tracking of camouflaged and occluded targets," in *Proc. of IEEE Workshop on Visual Surveillance, pp. 48-55*, 1999.
[5] T. Ellis and M. Xu, "Object detection and tracking in an open and dynamic world," in *Proc. of the Second IEEE International Workshop on Performance Evaluation on Tracking and Surveillance (PETS'01)*, 2001.
[6] C. Stauffer and W. Grimson, "Learning patterns of activity using realtime tracking," in *IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 22, pp. 747-757*, 2000.
[7] L. Ya, A. Haizhou, and X. Guangyou, "Moving object detection and tracking based on background subtraction," in *Proc. of SPIE Object Detection, Classification, and Tracking Technologies, pp. 62-66*, 2001.
[8] L. Li, W. Huang, I. Gu, and Q. Tian, "Foreground object detection from videos containing complex background," in *Proc. of the Eleventh ACM International Conference on Multimedia (MULTIMEDIA'03), pp. 2-10*, 2003.
[9] ——, "Statistical modeling of complex backgrounds for foreground object detection," in *IEEE Trans. Image Processing, vol. 13, pp. 1459-1472*, 2004.

[10] D. Socek, D. Culibrk, O. Marques, H. Kalva, and B. Furht, "A hybrid color-based foreground object detection method for automated marine surveillance," in *Proc. of the Advanced Concepts for Intelligent Vision Systems Conference (ACIVS 2005)*, 2005.
[11] D. Culibrk, O. Marques, D. Socek. H. Kalva. B. Furht. "Neural network approach to background modeling for video object segmentation," *IEEE Transactions on Neural Networks*, vol. 18, no. 6, pp. 1614–1627, Nov. 2007.
[12] E. Parzen, "On estimation of a probability density function and mode," *Ann. Math. Stat.*, vol. 33, pp. 1065–1076, 1962.
[13] D. Culibrk, D. Socek, O. Marques, and B. Furht, "Automatic kernel width selection for neural network based video object segmentation," in *International Conference on Computer Vision Theory and Applications (VISAPP 2)*, 2007, pp. 472–479.
[14] A. Prati, I. Mikic, M. Trivedi, and R. Cucchiara, "Detecting moving shadows: Algorithms and evaluation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, pp. 918–923, 2003.
[15] G. P. A. L. G.S.K. Fung, N.H.C. Yung, "Effective moving cast shadow detection for monocular color imagesequences," in *Proc. 11th International Conference on Image Analysis and Processing*, 2001, pp. 404– 409.
[16] U. G. P. O. Schreer, I. Feldmann, "Fast and robust shadow detection in videoconference applications," in *Proc. 4th EURASIP-IEEE Region 8 International Symposium on Video/Image Processing and Multimedia Communications (VIPromCom)*, 2002, pp. 371–375.
[17] M. P. A. P. R. Cucchiara, C. Grana, "Detecting objects, shadows and ghosts in video streams byexploiting color and motion information," in *Proc. 11th International Conference on Image Analysis and Processing*, 2001, pp. 360–365.
[18] D. Grest, J.-M. Frahm, and R. Koch, "A color similarity measure for robust shadow removal in real time," in *Proc. Vision, Modeling, and Visualization Conference (VMV 2003)*, 2003, pp. 253–260.
[19] A. Klaus, M. Sormann, and K. Karner, "Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure," in *Proc. 18th International Conference on Pattern Recognition (ICPR'06)*, vol. 3, 2006, pp. 15–18.
[20] C. Zitnick and S. Kang, "Stereo for image-based rendering using image over-segmentation," *International Journal of Computer Vision*, vol. 75, no. 1, pp. 49–65, October 2007.
[21] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 7–42, pp. 49–65, April-June 2002.
[22] S. Birchfield and C. Tomasi, "Depth discontinuities by pixel-to-pixel stereo," *International Journal of Computer Vision*, vol. 35, no. 3, pp. 269–293, December 1999.
[23] ISO/IEC 15938-3, Information technology - Multimedia content description interface; Part 3: Visual, 2002.
[24] Y. Deng, B. Manjunath, C. Kenney, M. Moore, and H. Shin, "An efficient color representation for image retrieval," *IEEE Transactions on Image Processing*, vol. 10, no. 1, pp. 140–147, January 2001.
[25] D. E. Knuth, *Art of Computer Programming, Volume 2: Seminumerical Algorithms, 3rd Edition*. Addison Wesley Professional, 1997.

[26] T. E. A. Cavallaro, O. Steiger, "Tracking video objects in cluttered background," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 4, pp. 575–584, April 2005.

4.5.2 Design and Implementation of an Optical Flow-based Autonomous Video Surveillance System

This section presents the design of a surveillance system based on optical flow. Considerations of the capabilities, limitations, and possible solutions to these limitations are presented. Additionally, an evaluation of the performance of optical flow in situations such as depth estimation, rigid classification, non-rigid classification, segmentation, and tracking will be presented. Our main contribution is a new system level architecture based on one algorithm for an entire video processing system. The case study is a video surveillance system, whereas optical flow is the main core.

4.5.2.1 Introduction

Video surveillance systems are often designed as modular systems composed of a variety of functional blocks such as motion detection, object tracking, depth estimation and object behavioral analysis [1, 4-7]. Adaptive systems, Kalman Filters, neural networks, and several image and video processing algorithms [2, 5] are being used without consideration of the impact of their computational performance on the entire system. Every component is connected in a serial configuration where the intermediate results of each block are not shared [9]. We focus on creating a more integrated video surveillance system based on one algorithm. We demonstrate that is feasible to share information between processing blocks, given a new alternative into the system level design scope.

A fundamental problem in the processing of image sequences is the measurement of optical flow (or image velocity). Once computed, the measurement of image velocity can be used for a wide variety of tasks ranging from passive scene interpretation to autonomous, active exploration [8]. Optical flow calculates disparities and depth estimation. Segmentation algorithms based on optical flow; objects matching and tracking are some examples of the optical flow potential [2, 3]. Therefore, optical flow could be considered as a strong candidate to demonstrate the performance for a surveillance system based on a single algorithm (as opposed to a combination of different functional blocks). Performance will be measured with different videos under different circumstances; objects classification, occlusion analysis and depth estimation will be part of testing. Although other algorithms could have been used instead of optical flow, the main contribution to the state-of-the-art is use of a single technique to reduce hardware resources and processing time while maintaining acceptable surveillance performance and quality.

Restricting our system for working only with optical flow information guide us for new alternatives for handling segmentation, depth estimation, tracking, and object classification. Implementation is always realized using just one camera which leads creative solution for solving problems such as depth estimation which is usually implemented with stereopsis techniques. We emphasize that the framework is the optical flow information, whereas the goal is the implementation of an entire video

surveillance system which demonstrates the importance of new system level architectures.

4.5.2.2 Proposed Method

This section describes the design and implementation of an autonomous video surveillance system based on optical flow calculations.

Figure 4.5.2.1 shows a high-level block diagram of the proposed system. The *optical flow (OF) calculation* block is highlighted to indicate that its results will drive several other blocks. A detailed explanation of each block follows.

Notice how the information is shared at the system level. Optical flow supply information for every module and also data is shared among different sub-blocks. This system architecture is designed for reducing the number of implemented algorithm and increasing the sharing data among modules.

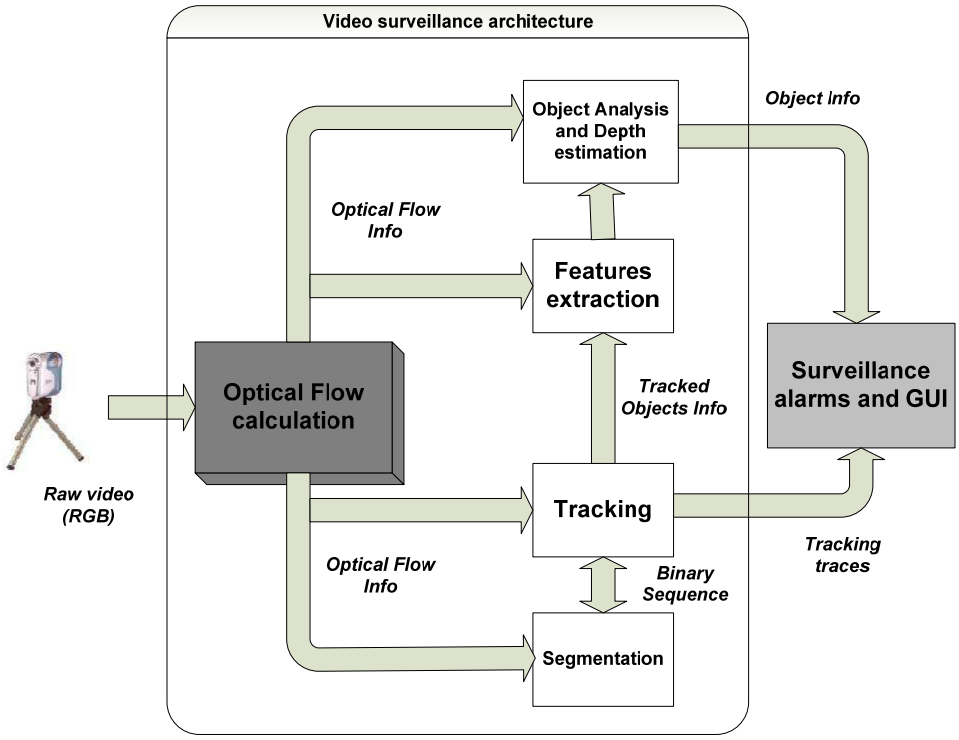


Figure 4.5.2.1. The architecture of the proposed system.

4.5.2.3 Optical Flow Calculation

The most common optical flow algorithms are listed below:

- Lucas-Kanade (Local method) [10]
- Horn-Schunck (Global method) [9]
- Optical flow using window pixels correlation [11]

Exhaustive research about optical flow computation has been done in the recent years. According to [8] Lucas-Kanade algorithm is chosen as the optical flow algorithm for this implementation. It has the best processing time with an acceptable quality.

4.5.2.4 Segmentation

The *Segmentation* block uses the results of the OF calculations to determine which pixels in the frame belong to the foreground and which pixels belong to the background. The threshold is a dynamic parameter and its value may change from one frame to the next due to several factors, such as: weather conditions, illumination or camera setting changes.

Figure 4.5.2.2 shows the average of optical flow vector lengths per frame. From frame 1 to 30 there is only background and average is close to 0.04. After frame 30 moving objects appear into the scene average increases. This fact is used for thresholding background and foreground optical flow vectors.

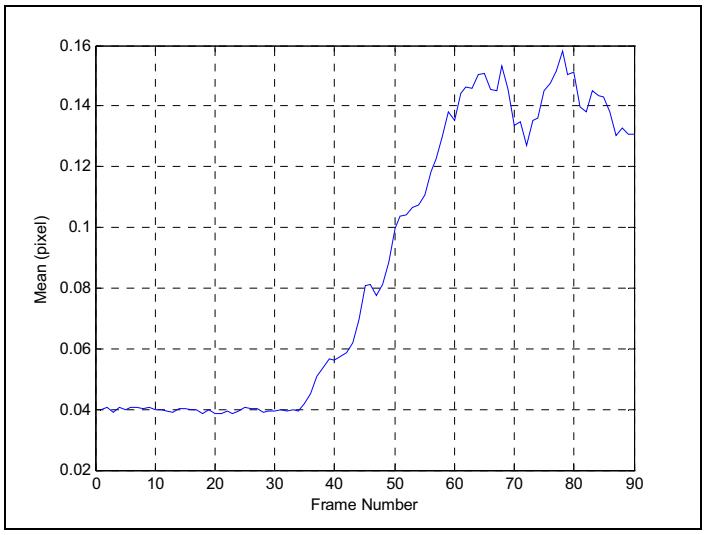


Figure 4.5.2.2. Average OF vector magnitude per frame.

The main challenge within this block is handling small OF vectors, which may be associated with background noise or relevant objects moving slowly between one frame and the next. We solved this problem by following the steps below:

1. Using a few initial frames with only background information, we calculate an initial threshold that can be used to suppress the optical flow noise. Threshold T is calculated assuming Gaussian noise with mean and standard deviation of the OF vector sizes (μ_B, σ_B), and a percentage of noise elimination P (Usually greater than 99%). At this point is important to mention that T is not an efficient value; it is higher than normal in order to eliminate background noise; affecting short-length foreground object vectors and reducing object contour accuracy.

2. Although several foreground object vectors were eliminated we have the number of real moving objects N for tracking. Now, we reduce T to $T_1 = T - \tau$

and then we count the number of objects N_1. If $N_1 = N$ then T_1 can be reduced again.

3. This process is repeated until the number of objects N_i at iteration i will be different of N.

4.5.2.5 Tracking

In our system each object – as determined by the *Segmentation* block – is represented by a bounding box and the coordinates of the object's centroid. Optical flow vectors are used as an object feature for tracking. Objects' trajectories are represented by centroid displacement.

The proposed tracking method is based on the optical flow tracking algorithm described in [12]. The main steps involved in determining where an object went are as follows:

1. Standard deviation σ and mean μ of the optical flow vectors surrounded by a bounding box in frame $k - 1$ are calculated and stored.
2. The bounding box in the frame $k - 1$ is shifted by the mean flow of that frame. This shifted window is called *the prediction window*.
3. Objects whose centroid is surrounded by the prediction window in frame k are going be part of the tracked object.
4. Bounding box, standard deviation σ and mean μ are calculated again for frame k.
5. Steps 2-4 are repeated until end of video sequence.

Internal data structures keep track of two types of objects: the ones which have been tracked for more than p consecutive frames (which are called *ActiveObjects*) and the ones which have not (*BufferObjects*). These data structures are updated on a frame-by-frame basis, instantiating new objects and promoting objects from the *BufferObjects* to the *ActiveObjects* category. The value of p must be chosen so that noise-like objects will not be tracked. For visualization purposes, objects currently being tracked are enclosed by a bounding box and their trajectory is painted on the screen (see Figure 4.5.2.4).

Occlusion is handled entirely within the tracking process. It is processed as follows:

1. Loading information of locations and objects that have disappeared and are not in the borders.
2. Alarming when a new object has appeared and it is not is the border.
3. Matching (Optical Flow features and aspect radio) previous lost ones with new no-borders objects.

If an object was close to the place where previous object disappear then it will be treated as stopped instead of occluded. We have mentioned that still objects could not be tracked due of the motion-requirement nature of OF. Thus, stopped objects are handled in the tracking block; we will show that this extra computation is not affecting the time processing of the tracking subsystem.

4.5.2.6 Feature Extraction

Feature extraction has been implemented using OF-derived features only. This is consistent with the philosophy of this system, which could be stated as "optical flow… and little else". Consequently, we currently use only the following features to represent each object:

- The mean value and the standard deviation of the x and y components of all the OF vectors associated with the object per frame.
- The number of OF vectors per object per frame.
- The aspect ratio of the object's bounding box per frame.

The modular nature of our architecture enables other, richer features (e.g., shape-, texture, and/or color-based) to be incorporated in the future if needed.

4.5.2.7 Object Analysis

A typical required step in behavior analysis is to classify objects as rigid and non-rigid. This can be achieved using optical flow under the assumption that OF vectors contained within the bounding box of a rigid object are likely to be parallel the object's theoretical displacement vector, whereas for non-rigid objects this will not hold.

We calculate the mean of the horizontal and vertical OF vector to help us determine, not only know the direction of an object, but also the oscillatory behavior of non-rigid objects. We also calculate the OF vector's standard deviation to determine how far OF vector components of the objects are from the theoretical object displacement vector: the smaller the standard deviation, the more likely to be a rigid object.

4.5.2.8 Depth Estimation

The number of OF vectors per object diminishes when the object moves away from the camera. This fact – in connection with the mean x and y OF values per object –is used to estimate the 3D trajectory of an object. Although calibration parameters are not taken into account; this first approach will show that this non-reference depth estimation is useful for:

- Handling group of objects merging at a specific point.
- Alarming if an object is in a possible non-permitted zone.
- Estimating object trajectory.

The process for non-reference depth estimation is depicted as follow:

- Number of pixels and mean value of the x and y components of all the OF vectors associated with the object is calculated per frame.
- Digital filtering is applied to previous information along every frame in which the object appear in order to remove non-rigid oscillations.

Displacement information is calculated by integrating object velocities; then different view plots and gray scale depth maps are stored for future analysis.

4.5.2.9 Experiments and Results

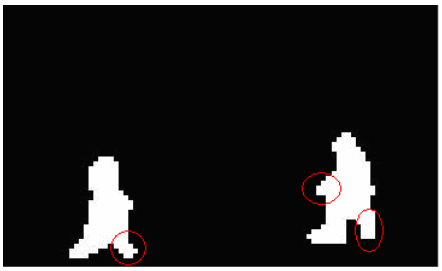
The system is implemented in MATLAB with a GUI as an interactive way to play with the code. All the sequences were taken in different scenarios with only one camera. Frame resolution is 320x240 pixels with RGB color information.

Segmentation:

Figure 4.5.2.3 shows optical flow vectors using a Lucas-Kanade algorithm with a window of 4x4 pixels. The original segmentation and corrected segmentation is also shown. Notice that partial parts of arms and legs are reconstructed after corrected segmentation algorithm.

(a) Optical flow

(b) Original Segmentation



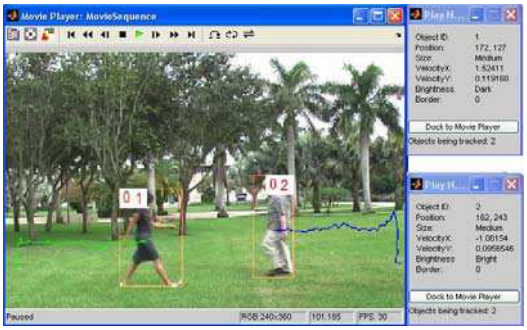
(c) Corrected segmentation

Figure 4.5.2.3. Segmentation process.

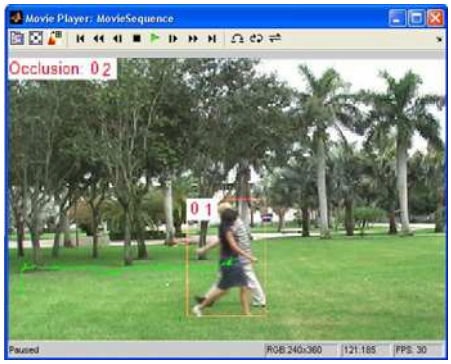
Tracking:

The way we proved the tracking system quality is by measure successful occlusions which are in a tracking system a very tough issue. Every object in our system has a

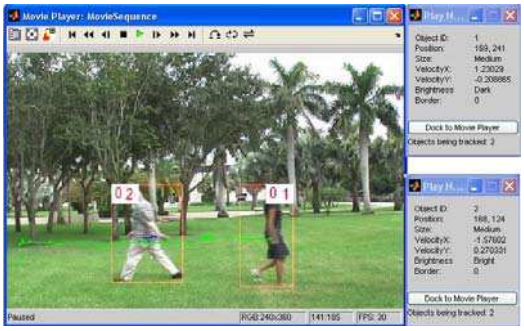
unique ID number, and then a successful occlusion occurs when the ID number is kept by objects after occlusion.



(a) Before Occlusion



(b) Occlusion



(c) After Occlusion

Figure 4.5.2.4. Object tracking in the presence of occlusion.

Figure 4.5.2.4 shows that a given object retains its ID number before (Figure 4.5.2.2(a)) and after (Figure 4.5.2.2(c)) the occlusion. Figure 4.5.2.2(b) shows and alarm indicating that the object with ID = 2 was occluded. Relevant information of each object (Object ID, Position, Size, Velocities, Brightness, and Flag's border) is displayed in the right-side windows.

Classification:

Figure 4.5.2.5 and Figure 4.5.2.6 show the rigid and non-rigid behavior of a human and a car for vertical component and number of pixels of optical flow. Notice the

oscillatory behavior of human signals which is reflected in standard deviation measures.

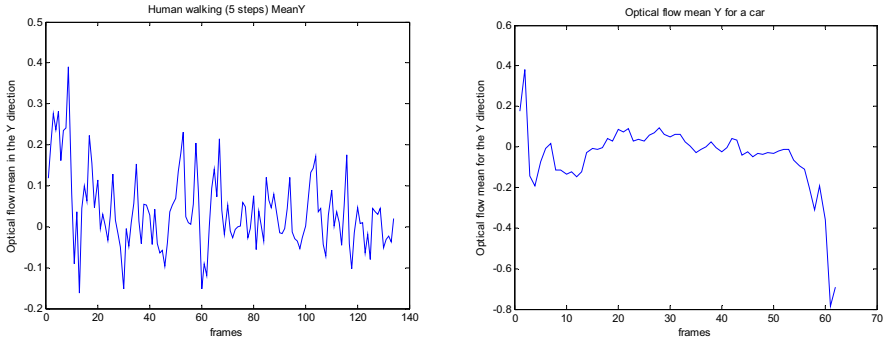


Figure 4.5.2.5. Mean vertical components for a car and human.

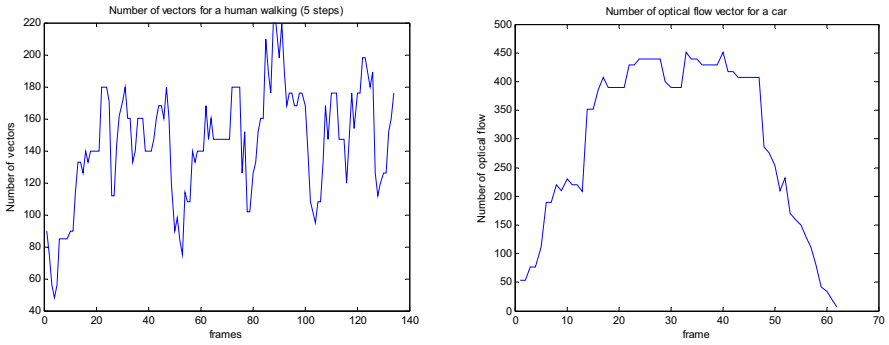


Figure 4.5.2.6. Mean number of optical flow vectors for a car and human.

50 car and human samples were taken and classified by using J48 tree classifier in weak.

Table 4.5.2.1. Object classification (standard deviation).

	TP rate	FP rate	Precision	Recall
Human	1	0.125	0.909	1
Car	0.875	0	1	0.875

Calculating the aspect ratio with the optical flow vector positions and classifying previous objects we got:

Table 4.5.2.2. Object classification (standard deviation and aspect ratio).

	TP rate	FP rate	Precision	Recall
Human	0.9	0	1	0.9
Car	1	0.1	0.889	1

Table 4.5.2.1 and 4.5.2.2 show object classification using one (standard deviation) and two (standard deviation and aspect ratio) measures.

Depth Estimation:

For a non-reference depth estimation of the trajectory shown in Figure 4.5.2.7 we use information of Figure 4.5.2.8 and 9. Information of Figure 4.5.2.8 let us describe vertical and horizontal real trajectory based on the average of OF vertical and horizontal components for the objects, respectively.

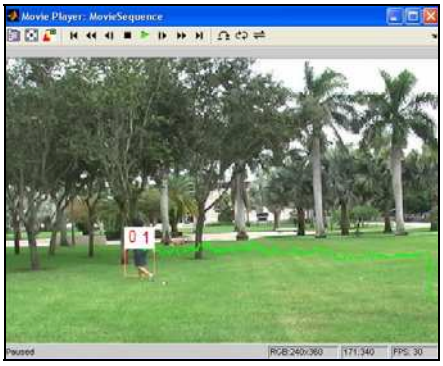


Figure 4.5.2.7. Object trajectory for depth estimation.

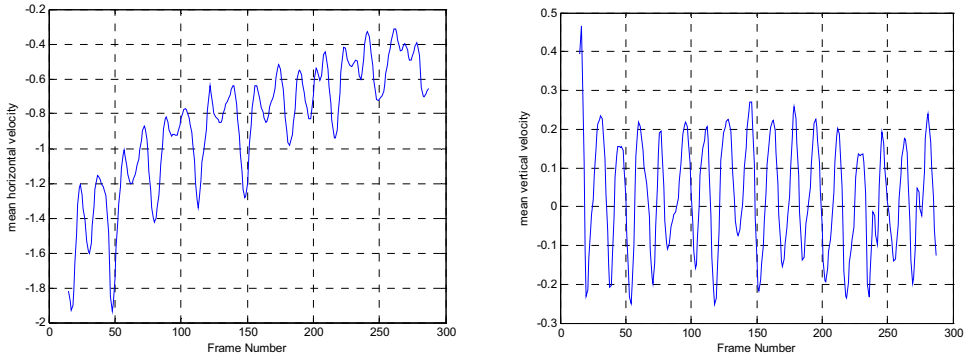


Figure 4.5.2.8. Average of OF Vertical and Horizontal components per frame.

Depth trajectory is calculated using information of Figure 4.5.2.9. The number of optical flow vectors for the object determine how close is the objects to the camera. The farther the object is, the greater the number of optical flow vectors we have.

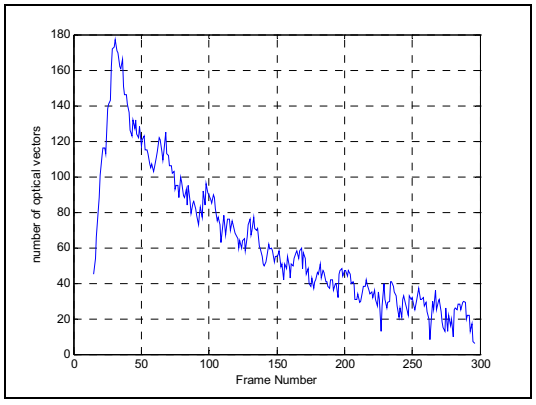


Figure 4.5.2.9. OF number of pixels per frame.

Previous information is used to determine the non-reference depth estimation as it is shown in Figure 4.5.2.10 in a gray scale depth plot.

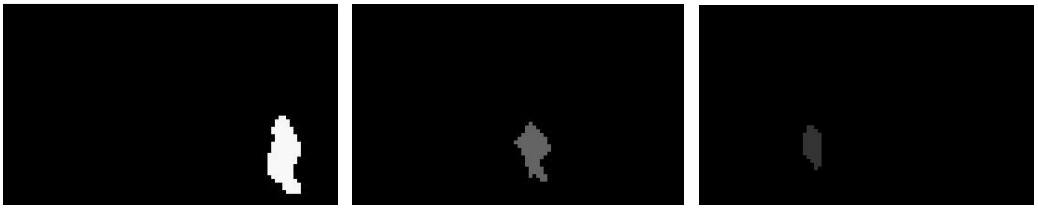


Figure 4.5.2.10. Relative depth map: frame 40 (left), frame 100 (middle), and frame 160 (right).

4.5.2.10 Concluding Remarks

A limitation of this system is the fact that if there is no optical flow there is no movement. It means that if an object stops trace will be loose. This system detects and warning when and object is not close to the borders and disappears. Relevant conclusions are listed below:

1. We have shown that a surveillance system can be designed by using a single technique (optical flow) with an acceptable performance and also with the possibility to be notoriously improved. 90% of object classification precision; more than 80% of occlusion success; depth estimation proposal with a single moving camera based on optical flow are relevant quality features for this system.
2. An algorithm for reconstructing segmented images is proposed for solving optical flow thresholding. Improvements such as faster algorithms and consecutive frames subtraction for reducing segmentation processing is part of the future work for segmentation.
3. Occlusion problems and stopped objects are handled in the tracking system. Optical flow problem with no movement could be solve in the tracking system by noticing where and object appear surprisingly.
4. A novel approach for depth estimation using optical flow and a single camera is proposed at this paper. Camera calibration for real depth measure is planned for future work.
5. Objects are classified as rigid and non-rigid using optical flow information. An oscillatory tendency leads notorious feature work for behavioral analysis of non-rigid object such as humans.

References for Section 4.5.2

[1] R. Collins, A. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, and O. Hasegawa, “A system for video surveillance and monitoring: VSAM final report,” *Robotics Inst., CMU-RI-TR-00-12*. 2000.

[2] W. Hu, T. Tan, L. Wang, and S. Maybank, "Survey on Visual Surveillance of Object Motion and Behaviors", *IEEE Transactions on systems and cybernetics*, Vol. 34, pp. 334-353. August 2004.
[3] Gilad Avid, "Determining three-dimensional motion and structure from optical flow generated by several moving objects". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 7, pp. 384-401. July 1985
[4] T. Kanade, R. Collins, A. Lipton, P. Burt, and L. Wixson, "Advances in cooperative multisensor video surveillance", In *Proceedings of the 1998 DARPA Image Understanding Workshop*, Vol. 1, pp. 3–24. November 1998.
[5] A. Yilmaz, O. Javed, and M. Shah, "Object Tracking: A Survey", *ACM computing surveys*, Vol. 38, No. 4, pp. 1-45. December 2006.
[6] D. Duque, H. Santos, and P. Cortez, "The OBSERVER: An Intelligent and Automated Video Surveillance System," *Proceedings of the International Conference on Image Analysis and Recognition*, pp. 989-909. 2006.
[7] I. Haritaoglu, D. Harwood, and L. Davis, "W4: Real-time surveillance of people and their activities," *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 22, pp. 809–830. Aug. 2000.
[8] J. Barron, D. Fleet, and S. Beauchemin, "Performance of Optical Flow Techniques", *Int. J. Comput.Vis*, Vol. 12, pp. 42–77. 1994.
[9] B. Horn and B. Schunck, "Determining Optical Flow", *Artificial Intelligence*, Vol. 17, pp. 185-203, 1981.
[10] B. Lucas, and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision", *Proc. DARPA IU Workshop*, pp. 121-130. 1981.
[11] W. Pratt, "Correlation Techniques of Image Registration", *IEEE Trans. Aerospace and electronic system*, Vol. 10, No. 3, pp. 353-358. May 1974.
[12] H. Tsutsui, J. Miura, and Y. Shirai, "Optical flow-based person tracking by multiple cameras", in *Proc. IEEE Conf. Multisensor Fusion and Integration in Intelligent Systems*, pp. 91–96. 2001.

4.5.3 A Model for Detecting and Tracking Humans Using Appearance, Shape, and Motion

In this section, we tackle the human detection problem as a classification task: moving foreground objects are classified as either human or not. The classification approach presented in this work is based on motion (periodic motion detection), appearance (skin color detection) and shape (MPEG-7 shape descriptors). A modular infrastructure for data collection, object instantiation and tracking was also implemented and can be expanded in future related work.

4.5.3.1 Introduction

In a security environment, objects by themselves are not the initiators of a special circumstance; it is the interrelation between humans and objects, or just between humans, that can give hints to proactively identify an anomaly. Hence, human detection must be the base in systems where we need to extract even higher level information, such as recognizing people's activities.

4.5.3.2 Periodic Motion Detection

If we observe walking or running people, we may notice that this type of motion contains periodic characteristics. This was exploited by Cutler et al [1] and others [2-5]. Another technique that falls under this category is the one proposed by Lipton [6], who uses residual flow to analyze periodicity and rigid motion.

There is behavioral evidence that animals and humans can recognize biological motion according to its periodic characteristics [7].

In the technique proposed by [1] we need the isolated image of an object across N consecutive frames. Once we accumulate that information, we resize the different images according to the median values. Then we calculate its correlation matrix, according to the following equation:

$$S_{t_1,t_2} = \sum_{(x,y)\in B_{t_1}} |O_{t_1}(x,y) \quad O_{t_2}(x,y)|$$

Where B is the bounding box of object O, and t_i makes reference to the different resized instances of the object ($0 \leq i \leq N$). The next step is the computation of the correlation matrix Discrete Fourier Transform (DFT). The object's motion will be considered as periodic if there are values that meet the condition below:

$$P \geq \mu_P + K\sigma_P$$

Where P is the DFT of the correlation matrix, K is a threshold value (typically 3 according to the authors), and μ_P and σ_P are the mean and standard deviation of P respectively.

4.5.3.3 Skin Color Detection

Motion-based classification techniques are useful when there are humans walking in the scene, but what if the present subjects are not moving? Using appearance-based classification, we can point to objects in the scene that contain determined human characteristics, such as color. According to [8], skin color is determined by a single melanin pigment, and only its density differs between different ethnic groups. This appearance feature can be used as one of the thresholds of our human detector.

In this scheme, we take advantage of the independence between luminance and the chroma components of the *YCrCb* color space. After performing background subtraction, the chroma components of the foreground pixels are compared against predefined thresholds of the skin color. If the values Cr and Cb fall within the range [133, 173] and [77, 127], the pixel is labeled as "skin color" [9].

4.5.3.4 Shape-Based Detection

If we use motion and appearance techniques in a scene where our actors are walking perpendicularly to the plane of the camera or are hiding their faces, our hypothetical system, most likely, wouldn't be able to recognize them as humans. To circumvent these limitations, we can use shape characteristics of the human silhouette in order to detect humans in the scene. In this work we use two such techniques: dispersedness and MPEG-7 region-based shape descriptor.

Dispersedness

The mathematical definition for this metric is [10]:

$$Dispersedness = \frac{Perimeter^2}{Area}$$

where *Perimeter* is the number of pixels that belong to the contour of a shape, and *Area* the number of pixels contained inside the contour.

Humans tend to have more complex shapes than vehicles. If we compare a vehicle and a human in an image where both silhouettes are clearly defined, the human will have greater dispersedness due to the greater complexity of its shape. The computational cost of this metric is low.

MPEG-7 Region-based Shape Descriptor

MPEG-7 is a standard specifically designed for multimedia content description. In this section we refer to the region-based shape descriptor, which in order to describe a shape uses the Angular Radial Transform (ART) [11] to extract the set of coefficients [12] that define the descriptor.

The distance (or dissimilarity) between two shapes described by the ART descriptor is calculated using the function:

$$Dissimilarity = \sum_i \left\| M_d[i] - M_q[i] \right\|$$

where M is the array of ART descriptor values of images d and q respectively;

The descriptor is characterized by its small size, fast extraction time and matching [13].

4.5.3.5 The Proposed Approach

The proposed solution arranges processing blocks following the diagram in Figure 4.5.3.1. The flow of data in the block diagram is explained in the next subsections:

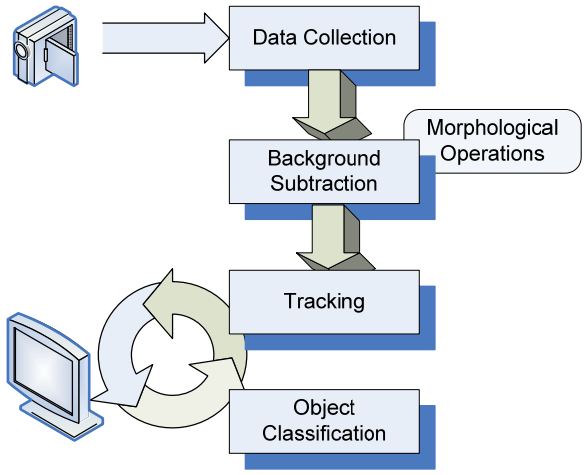


Figure 4.5.3.1. Block diagram of the proposed solution.

Data Collection

In this process we capture, edit, resample, and store the video sequences.

Background Subtraction

This block extracts the foreground objects present in the video sequence. The proposed solution uses the prototype described in [14]. The output of this block is a set of binary images with the same size as the original ones. This processing block is fundamental in this scheme, performance of subsequent blocks depend on it, therefore its output must be as noise free as possible. In this implementation a block of *Morphological Operations* is applied to the output of the *Background Subtraction* block in order to filter possible noise.

Tracking

At this point we are ready to apply the Tracking block that follows the displacements of foreground objects present in the scene. The solution is based on the work developed by [15]. The implemented tracking algorithm is based in the next parameters [16]:

- *Object representation:* object's bounding box.
- *Feature selection for tracking:* manual, foreground object's size.
- *Object detection:* background subtraction based.
- *Object Tracking:* point tracking.

We chose to model the tracking problem using the finite state machine (FSM) shown in Figure 4.5.3.2.

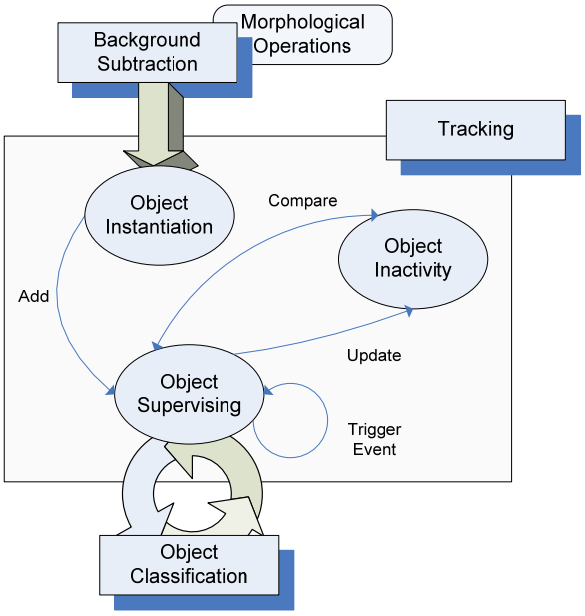


Figure 4.5.3.2. Monitoring state machine.

This machine works with each foreground object individually and in order to keep track of the object's related information each state has an associated list or other data structure. The transition of a foreground object from one state to another can be modeled also as the flow of information between different data structures. Figure 4.5.3.3 shows the associated data structures for the different states of the machine:

The 3-Frame Buffer belongs to the Object Instantiation state. The Objects and Active Tracking Lists belong to the Object Supervising state. The Inactive Objects List belongs to the Inactivity state. The Priority Queue belongs to the Trigger State action.

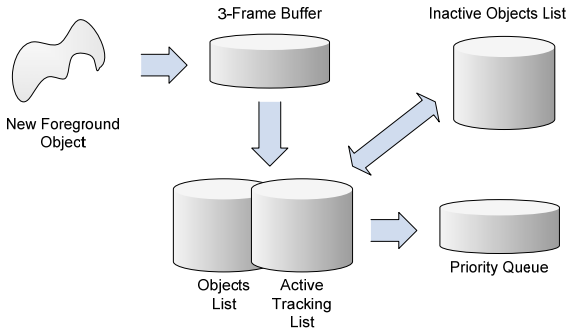


Figure 4.5.3.3. Associated data structures for the monitoring state machine.

Object Instantiation

In this state, we label the unconnected objects in the scene. Selective information about each object is obtained. For our particular needs, the algorithm extracts the centroid of the object, an index list of the pixels contained in the object, and the appropriate data needed to locate its bounding box. When a new object is discovered

is placed in a 3-frame buffer (see Figure 4.5.3.3), i.e., if the object is successfully tracked over three frames, it is then removed from the buffer and is ready to enter the Object Supervising state.

Object Supervising

In this state, an object's trajectory is calculated according to its size, present and previous bounding box. Also, for each object, the system keeps track of different features such as size, position, velocity, brightness level, etc.

The last function performed in this state is the comparison against the objects in the Inactivity state. If an object resembles one of the objects there, we have reasons to believe it is the same object, which had disappeared and now appears again. In that case we reuse a previous object accumulated data.

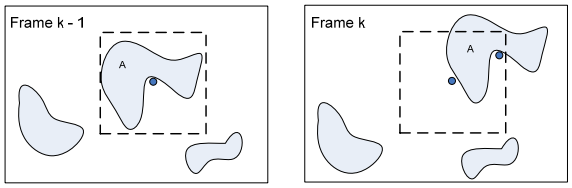


Figure 4.5.3.4. A foreground object's displacement.

Trigger Event

This is an action performed by the *Object Supervising* state. For this action, different events are placed in a priority queue in which each event's priority is determined by the results of the Object Classification block (see Figure 4.5.3.1). Events with higher priority will be those where our object's motion, appearance and shape are human-like, whereas other events with lower priority are those where an object moves from one state to the other. In this solution the event handler places the events in two different log files. The events with the highest priority are placed in the main log file and graphical alarms are generated for them. The other events are stored in the secondary log file.

Object Inactivity

An object will reach this state when it is not visible anymore. For this state we count with an Inactive objects list, where each element has a specific time to live (ttl). In this state, flags and logical comparisons are computed in order to handle occlusion or possible feature inheritance. In the example of Figure 4.5.3.5, Object *B* has been occluded by object *A,* therefore becomes inactive and its *insider* flag is raised because disappeared inside the dotted region of the frame.

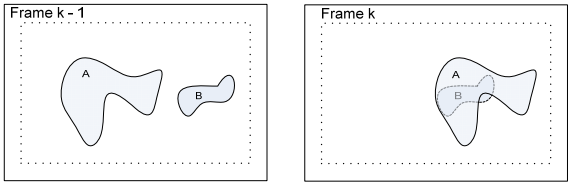


Figure 4.5.3.5. Occlusion example.

When the object *B* reappears inside the dotted region, it is labeled as an insider and it is compared only with the inactive insider objects and inherits the properties of the object whose size is the most similar to. In cases where the new object appeared outside the dotted region, it is considered as outsider and inherits the characteristics of the first inactive object that meets predetermined size and position conditions.

Object Classification

The techniques chosen in this block for human detection are periodic motion detection [1], skin color detection [9] and the MPEG-7 shape descriptors.

The periodic motion detection sub module requires accumulated data of a specific number of frames. When this information is complete, the algorithm is applied and different flags in the Active Tracking List and Objects list are raised or cleared. The *skin color detection* and *shape descriptors* sub modules outputs only require information from the actual frame (frame *k*), However, since the system has to wait for the *periodic motion detection* sub module output, temporal averages from their outputs are also stored. As future work, a voting strategy could be applied and the object will be labeled as human if more than one sub module voted in favor. If the results don't show anything "interesting", we continue supervising, otherwise the system will trigger an event. Naturally, at a certain point in time a tracked object could disappear from view. This condition will take the object to the inactivity state.

4.5.3.6 Experimental Results

Figure 4.5.3.6 shows the tracked trajectory for a human subject and its bounding box. In the right side of the picture we find the *trackingData* structure where we accumulate data for the tracked objects in the video sequence.

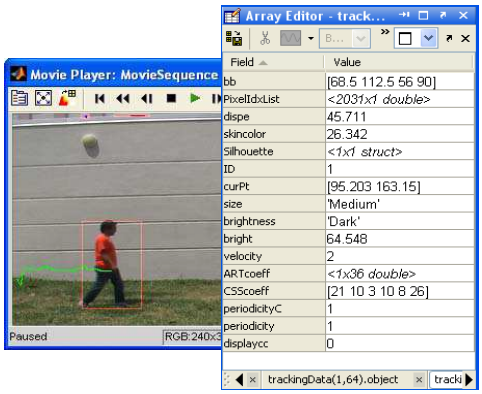


Figure 4.5.3.6. Trajectory, bounding box and data structure for tracked human subjects.

To determine if an object has come back to the scene, i.e. was inactive in the frame (*k-1*) and is active in the frame *k*, a parameter or set of parameters must be compared. In the current implementation the objects' position and size, are the parameters used to perform the comparison. In short, if the position and size conditions are met the new object inherits an inactive object's data.

Figure 4.5.3.7. Object inactivity and occlusion handling.

Figure 4.5.3.7 illustrates a scene where the previous conditions proved to be effective. In the scene, a walking human identified with ID 4(frame 1) is occluded by a vehicle (frame 2), becoming inactive. When the object reappears, first it is considered as a new object. A short while later the *compareInactive* function determines if it is similar to an inactive object. Since this is true in this case, it inherits its accumulated data (frame 3), e.g. the inactive objects' ID, the "Human" label, and the "trace" of previous tracking (green trace in Figure 4.5.3.7)

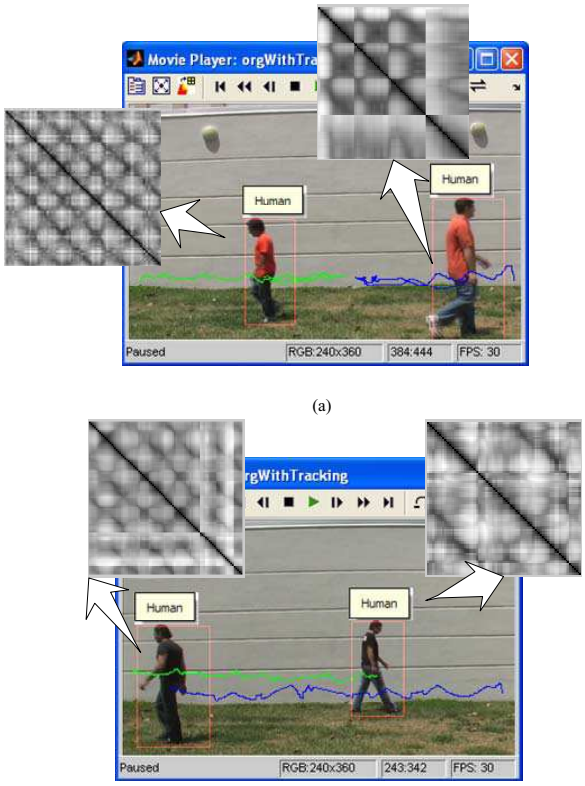


Figure 4.5.3.8. Human subjects and their respective correlation matrices: (a) scene one (b) scene two.

Figure 4.5.3.8 shows two different scenes where periodic motion was used to detect human subjects. In Figure 4.5.3.8 (b) the subject with the green trajectory walks at a lower pace in comparison with the other subject present in the scene. If we chose the same column of the correlation matrices, the fast Fourier Transform confirms that the main frequency components are different. According to this, better correlation matrices could be obtained if the number of frames used to create them, are calculated based on the objects' speeds.

In Figure 4.5.3.8 (a) the correlation matrix of the subject in the right side was calculated after changing his motion direction. This means that subject started walking from right to left and then turned around and walked from left to right. Once again, better results can be obtained if the motion direction is taken into account to calculate the correlation matrix,

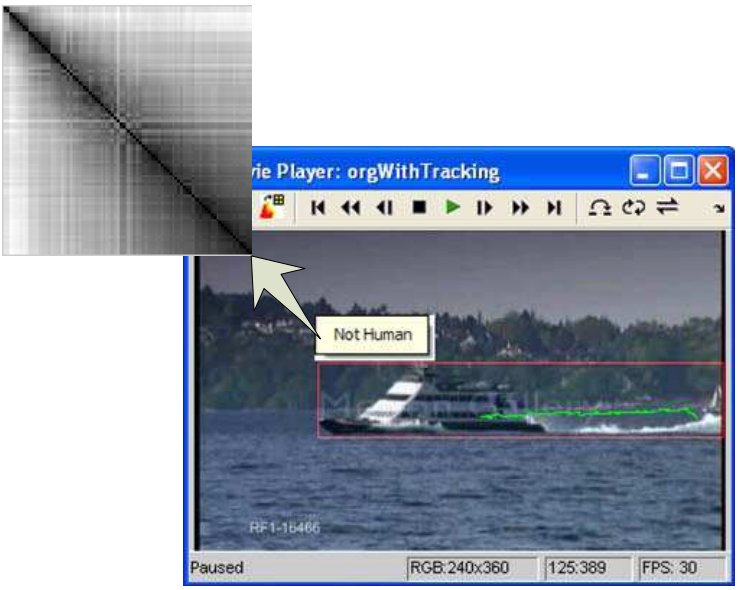


Figure 4.5.3.9. A rigid subject and its correlation matrix.

Figure 4.5.3.9 shows a rigid object and its correlation matrix. In comparison with the matrices obtained in Figure 4.5.3.8, the lack of symmetry to the main diagonal is perceptible, and therefore, the object is labeled as not-human.

Figure 4.5.3.10. Skin color detection on a sample scene.

Figure 4.5.3.10 shows the output of the skin detection function. From a qualitative point of view the algorithm is successful labeling skin pixels in foreground objects. However, the frame in the lower right side shows some false positives. The explanation to this: shadows are elected as part of the foreground object in the background subtraction process, also parts of the floor where the shadows occur are skin colored, and therefore labeled as skin pixels.

According to [**Error! Bookmark not defined.**] human shapes have greater dispersedness in comparison to vehicles; this is not necessarily true in low resolution video (128x160), as confirmed by the dispersedness values in Table 4.5.3.1.

Table 4.5.3.1. Dispersedness values for objects in a sample scene.

Object ID	Mean	Std	Car	Human
1	23.86	3.87		x
2	22.98	5.57	x	
3	17.06	2.81		x
4	30.06	13.54	x	
5	24.70	6.87	x	
6	24.10	6.37		x
7	23.85	7.37	x	
8	28.22	4.49		x
9	19.42	5.96		x
10	28.38	6.51	x	
11	28.21	8.79	x	
12	21.86	5.55	x	
13	20.14	2.58	x	

4.5.3.7 Conclusions

Through study and experimentation, this work has reached the following conclusions:

Different background subtraction techniques where implemented and compared. The one that showed better qualitative and quantitative results was the one implemented in [14]; as a downside the tuning of the alpha parameter proved to consume more resources than anticipated. The measure of computation time is not critical in this implementation, but must be improved if real time implementation is required.

It is important to highlight that the performance of the tracking block is strongly dependent on the outputs of the background subtraction block, Also, the comparison features used in this work (centroid and bounding box) proved to work in scenes with a limited number of subjects (maximum two). Use of new features and their probabilistic properties could improve this block's accuracy and computational performance when several objects are present in the scene.

Data collection from the MPEG-7 shape descriptors was accomplished, however further testing and new metrics are required to add these components to the human detection algorithm. For appearance the skin detection component proved to detect hot spots in ideal conditions; however is not reliable as a stand alone human detector in cases where there is not skin color at all or a subject's clothes are skin colored. For motion only one descriptor was implemented; use of an object's velocity and direction values could improve the performance of this sub module. Empirical evaluation determined that motion information was able to distinguish humans from non-humans.

References for Section 4.5.3

[1] R. Cutler and L. S. Davis, "Robust real-time periodic motion detection, analysis, and applications," IEEE Trans. Pattern Anal. Machine Intell., vol. 22, Aug. 2000, pp. 781–796.
[2] M. Allmen, "Image sequence description using spatiotemporal flow curves: toward motion-based recognition," PhD thesis, University of Wisconsin-Madison, 1991.
[3] Polana, R and Nelson, R C, "Detecting activities", Proc. Conf. on Computer Vision and Pattern Recognition. New York, NY, June 15-17 1993, pp 2-7
[4] Tsai, P-S, Shah, M, Keiter, K and Kasparis, T, "Cyclic motion detection for Motion Based Recognition," Pattern. Recognition, vol. 27, 1994.
[5] Allmen, M C and Dyer, C R 'Cyclic motion detection using spatiotemporal surfaces and curves', Proc. 10th Int. Conf. Pattern Recognition, Atlantic City, NJ (1990) pp 365-370
[6] A.J. Lipton, "Local application of optic flow to analyze rigid versus non-rigid motion," in Proc. Int. Conf. Computer Vision Workshop Frame-Rate Vision Corfu, Greece, 1999.
[7] G. Johansson, "Visual Perception of Biological Motion and a Model for its Analysis," Perception and Psychophysics, vol. 14, 1973, pp. 210-2 11.
[8] Sangho Park, J.K. Aggarwal, "Simultaneous tracking of multiple body parts of interacting persons," Computer Vision and Image Understanding,2006, pp. 1-21.
[9] Chai, D.; Ngan, K.N., "Face segmentation using skin-color map in videophone applications ," Circuits and Systems for Video Technology, IEEE Transactions on , vol.9, no.4, Jun 1999, pp.551-564.
[10] A.J. Lipton, H. Fujiyoshi, R.S. Patil, "Moving target classification and tracking from real-time video," Proceedings of the IEEE Workshop on Applications of Computer Vision, 1998, pp. 8–14.
[11] Miroslaw Bober, "MPEG-7 Visual Shape Descriptors," IEEE Transactions On Circuits And Systems For Video Technology, vol. 11, No. 6, Jun. 2001, pp 716-719.
[12] Miroslaw Bober, "MPEG-7 Visual Shape Descriptors," IEEE Transactions On Circuits And Systems For Video Technology, vol. 11, No. 6, Jun. 2001, pp 716-719.
[13] http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm#E12E30
[14] Dubravko Culibrk, Oge Marques, Daniel Socek, Hari Kalva, Borko Furht, "A neural network approach to bayesian background modeling for video object segmentation," VISAPP, 2006, pp. 474-479.
[15] Jeremy Jacob. "Motion Tracking In The Presence Of Dynamic Background Movement". Final report for the Video Processing course, FAU, 2005.
[16] A. Yilmaz, O. Javed, M. Shah, "Object Tracking: A Survey," ACM Computing Survey, vol. 38, 2006.

4.5.4 Using a Computational Model of Human Visual Attention for Detecting Objects in Images

Computational models of human visual attention describe the early processes of human vision by predicting the areas of an image that are likely points of fixation. In this work we analyze the suitability of a computational model of human bottom-up visual attention for detecting salient regions of interest in images. The performance of using the predicted salient points to detect regions of interest is evaluated. Our results suggest that the points of attention generated by such models provide a principled,

unsupervised, biologically-inspired method for extracting seeds which can be subsequently used by region growing segmentation algorithms.

4.5.4.1 Introduction

The human vision system is amazing in its efficiency. It can process an enormous amount of visual information in real-time. One key ability of the human vision system is attention, which prioritizes the areas of a scene the eye fixates on. Attention may be bottom-up, based on instinct and reflexes, or top-down, derived from past memories and the task at hand. Bottom-up attention occurs quickly, before top-down influences direct where the eye will fixate next.

By bringing the most important portions of a scene to the forefront of the human's vision processing task, bottom-up attention serves an important function in survival. In this work, we consider these bottom-up, salient points of attention as signs of likely locations of regions of interest which merit further processing.

This work occurs within the context of a complete image retrieval system. Points of attention can be used to generate surrounding regions of interest, either by seed-based region-growing, or by segmenting the image and using the points of attention to select the relevant segments. Then, features are extracted from the generated region and compared using an appropriate similarity measure. The scope of this work is the inspection of the appropriateness of using points of attention generated by the Itti-Koch computational model of bottom-up visual attention [6]. Using several datasets, we compare the ground truth regions (expressed as bounding boxes around the objects of interest within the image) with the points of visual attention generated by the computational model to determine the performance of using these points as seeds for later processing.

4.5.4.2 Scope

In the overall project, a content-based method of retrieving images based on their salient regions of interest, not on their global properties, is proposed. Consider the image retrieval task of searching for a tennis ball. While tennis balls generally have a consistent appearance, the global properties of the images they appear in tend to vary (e.g. against a sky, a grass court, a clay court, etc.). By extracting features from the salient regions of interest rather than the entire image more relevant results may be obtained. Specifically, in the image retrieval task of searching for objects within images, salient points provide important indications of where these objects are located.

Figure 4.5.4.1 summarizes the proposed framework. There are three main phases. First, a computational model of visual attention is used to model early vision. Its output are points of fixation which are intended to correspond to predicted salient regions in the image. The combination of these points and the original images are provided to a seed-based region-growing module which produces masks for each computed region. Finally, features are extracted from the areas of the original images distinguished by region masks. These feature vectors can be used in a variety of CBIR applications such as querying or clustering.

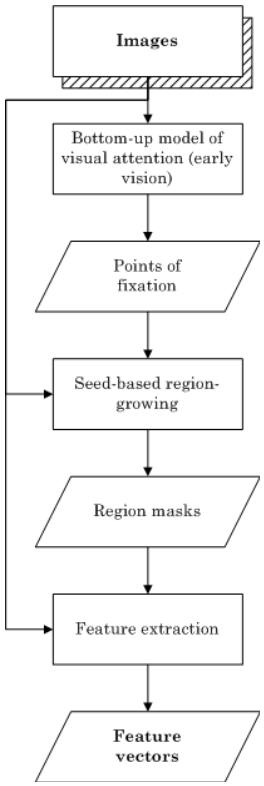


Figure 4.5.4.1. Illustrated in this figure is the proposed framework, of which this paper describes one component, the *Points of fixation* block.

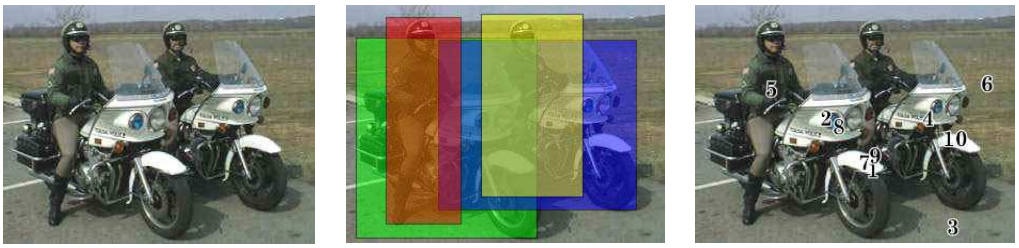
4.5.4.3 Dataset

Seven databases comprise the dataset used in this work. The selected databases are summarized in Table 4.5.4.1. In this table the following metrics are considered:

- *Images*: the number of images in the database.
- *Objects*: the total number of annotated objects. Because each image has, on average, more than one annotated object, the number of annotated objects is greater than the number of annotated images.
- *Classes*: the number of different object categories in the dataset. The number of classes must be considered alongside the nature of those categories, not in isolation. Overlapping categories (e.g. ``cows" and ``farms") are more difficult to distinguish between than disjoint categories (e.g. ``people" and ``airplanes"). Several classes used in the databases in this work are ``people", ``motorcycles", and ``cows".
- *Density*: the total number of annotated objects in the database divided by the total number of annotated images in the database.

We only considered fully-annotated databases where each image has at least one annotated object. Additionally, only databases with manually annotated objects were included. This annotation can be either bounding boxes surrounding the regions of interest (Figure 4.5.4.2 (b)), pixel-wise image masks (Figure 4.5.4.3 (b)), or polygons drawn around relevant objects in the image. In this work all ground truth annotation

was either originally provided in bounding box form or converted to bounding boxes surrounding ground truth masks, for fairness in the results.



(a) (b) (c)

Figure 4.5.4.2. Ground truth bounding boxes and the calculated points of attention (original image from [4]). (a) is the original image, (b) are the ground truth bounding boxes, and (c) are the calculated points of attention.

(a) (b) (c)

Figure 4.5.4.3. Object masks and the calculated points of attention (original image from from [5]). (a) is the original image, (b) are the ground truth masks, and (c) are the calculated points of attention.

Table 4.5.4.1. Image databases.

Name	Images	Objects	Classes	Density
STIMcoke **Error! Reference source not found.**	104	104	1	1.00
VOC2005 1 **Error! Reference source not found.**	1578	2209	4	1.40
VOC2005 2 **Error! Reference source not found.**	654	1293	4	1.98
VOC2006 Trainval **Error! Reference source not found.**	2618	5455	10	2.08
VOC2006 Test **Error! Reference source not found.**	2686	4052	10	1.51
VOC2007 Trainval **Error! Reference source not found.**	9963	24640	20	2.47
VOC2007 Test **Error! Reference source not found.**	4953	15662	20	3.16
Arithmetic mean	3222.29	7630.71	9.86	1.94

The STIMcoke database [5] was designed to have one salient region of interest (a soda can) within each image and was selected to provide a more controllable reference case (due to its small size, limited number of regions, and salient properties of the objects of interest). The PASCAL Visual Object Classes (VOC) [4] Challenge aims to classify objects in realistic scenes. An annual competition is held to compare various image retrieval systems. The selected databases contain images in categories such as people, cars, and bicycles. One benefit of using the VOC databases for experiments is that they are completely annotated. The VOC2005 databases are less complex (and challenging) than the ones used in the more recent challenges in every respect. The VOC2007 database is particular challenging given the number of objects and the variety of classes.

4.5.4.3 Points of Attention

To evaluate the suitability of using the seeds generated by a computational model of visual attention as seed points for a region-growing algorithm we used the Itti-Koch model of visual attention [6]. Since its introduction a decade ago the Itti model has refined and used in a wide variety of applications. There are implementations of the model publicly available in several programming languages. It was employed to generate ten points of attention for each image in the dataset. The objective was to record a generous number of points of attention which would be predicted for times that are far beyond reasonable application of the model (times that exceed those of bottom-up attention). As the model is bottom-up, it was designed only to represent the first few saccades of the human vision system, before top-down factors begin to influence the vision task [6].

This work determines the one unknown parameter -- until what time should points of attention be considered? If left too long the points will be meaningless, as they exceed the design intention of the model. We can empirically determine until what time to utilize the model by finding the point in time at which the increase in the number of positive results is outpaced by the number of negative results. In other words, recall would be perfect if each pixel in the image was attended to, but they would also be meaningless as the number of negative results would be maximal as well. Determining a reasonable point in time to cutoff results will help make the system efficient. We used receiver operating characteristic (ROC) curves [1] to make this assessment.

4.5.4.4 Experiments

A publicly available MATLAB implementation of the Itti-Koch model was used to generate the points of attention [7]. Ground truth was available for each image in the form of one or more bounding boxes for target objects and a label associated with each bounding box. Each generated point of attention has a set of coordinates and a time (in milliseconds, the predicted time of the fixation). There are two perspectives from which the results can be calculated, from the point of view of the points of attention of from the perspective of the ground truth regions. For the former (points of attention):

- *True positive*: this is the ideal case, when a predicted point of attention falls within a ground truth bounding box for an object within the image. It is denoted as TP_{poa}. In Figure 4.5.4.4 points 2, 3, 4, and 5 are all true positives. Point 5 is counted only as a single true positive.
- *False positive*: also known as *Type I error* or α *error*. This occurs when a point falls outside of any of the bounding boxes in an image, incorrectly identifying a region of the image as being an object. It is denoted as FP_{poa}. Point 1 in Figure 4.5.4.4 is a false positive as it is a predicted point of attention that is not within any bounding box.

False negatives and true negatives do not apply to points of attention. For the latter (regions of interest):

- *True positive*: a region that is hit by a point of attention is a true positive. A region can only count once (subsequent hits are ignored). It is denoted as TP_{roi}. In Figure 4.5.4.4 regions a, b, and d are true positives as they have all been hit be points of attention.
- *False negative*: also known as *Type II error* or β *error*. This error occurs when an ground truth object eludes all of the predicted points of attention and is never identified as a region for further inspection, and thus can never be recovered at a later stage of processing. It is denoted as FN_{roi}. In Figure 4.5.4.4 bounding box c is a false negative as it is never hit by a predicted point.

False positives and true negatives do not apply to regions of interest.

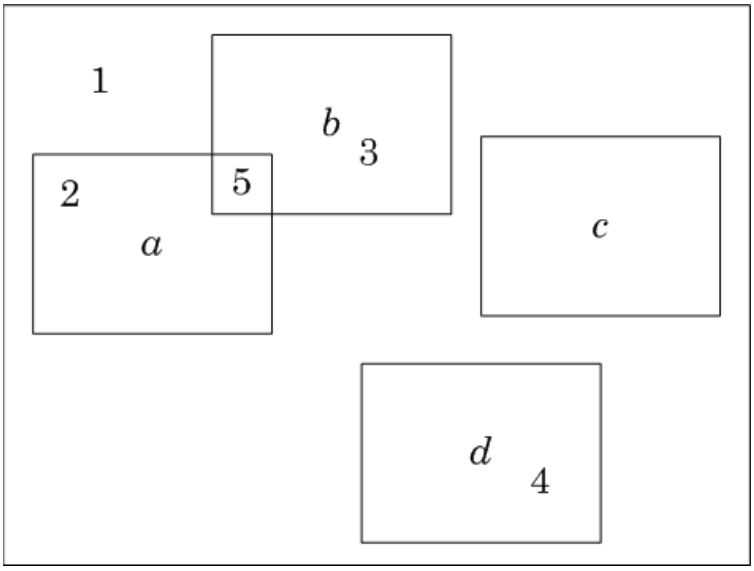


Figure 4.5.4.4. A sample of the scoring methodology is illustrated in this figure. Numbers 1-5 represent predicted points of attention while letters a through d indicate ground truth regions of interest.

Additionally, the maximum number of potential false positives ($MaxFP_{poa}$), defined as the sum of the number of true positives and false positives

$$MaxFP_{poa} = TP_{poa} + FP_{poa}$$

Samples were taken ever 100ms between 100ms and 10000ms. For each sample TP_{poa}, TP_{roi}, FP_{poa}, and FN_{roi} were recorded, based on where the points of attention landed relative to the ground truth. Two metrics were then calculated:

- *Hit Rate*: the proportion (in percent) of true positives from the maximum number of possible true positives
- *False Alarm Rate*: the proportion (in percent) of false positives from the maximum number of false positives

The Hit Rate is defined as follows:

$$HitRate_{roi} = \frac{TP_{roi}}{TP_{roi} + FN_{roi}}$$

The False Alarm rate is defined as follows:

$$FalseAlarmRate_{poa} = \frac{FP_{poa}}{MaxFP_{poa}}$$

4.5.4.5 Results and Discussion

Results are illustrated as a receiver operating characteristic (ROC) curve. An ROC curve plots the Hit Rate vs. the False Alarm Rate. In an ideal case, the ROC curve will be close to the top left corner of the plot, where the Hit Rate is high and the false positive rate is low. An inflection point on the ROC curve indicates a point where returns begin to diminish (i.e. beyond where the samples do not contribute to the results). The time associated with the sample at that inflection point is recorded.

For each database the point at which gains in the Hit Rate began to diminish with respect to increases in the False Alarm Rate was selected (Table 4.5.4.2). The associated time was recorded. This value was fairly consistent (ranging between 900ms and 1200ms with an arithmetic mean of 1085.71ms) regardless of the database. While the predicted times vary for each image (there is no set amount of points predicted by a certain time) this experimental result is well within the bounds of the data we collected (ten points of attention were generated per image and in nearly every case the tenth point was predicted for a time well beyond 10000ms). The relationship between the number of points and time could not be known before the experiments are executed but is implied by the generated ROC curves.

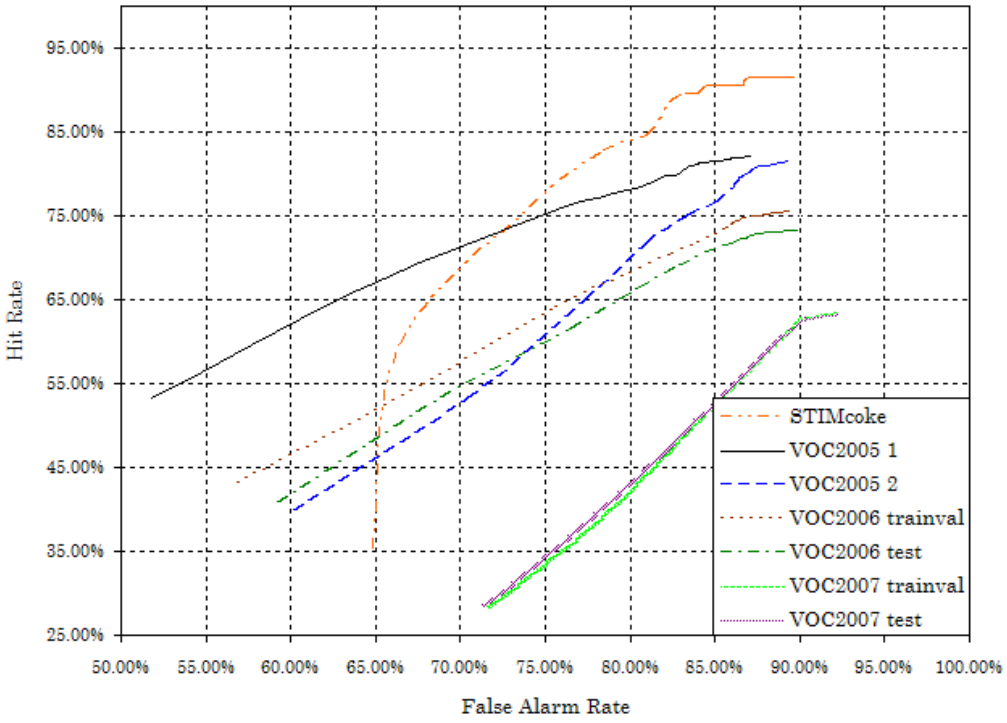


Figure 4.5.4.5. Receiver operating characteristic (ROC) curve displaying Hit Rate vs. False Alarm Rate for variances in the time parameter.

Table 4.5.4.2. Experiment results.

Database	Time (ms)	Hit Rate	False Alarm Rate	Occupied
STIMcoke [5]	900	90.38%	84.49%	1.70%
VOC2005 1 [4]	1200	94.57%	83.73%	38.50%
VOC2005 2 [4]	1000	92.27%	87.19%	36.48%
VOC2006 Trainval [3]	1000	90.72%	86.25%	44.61%
VOC2006 Test [3]	1100	89.95%	86.94%	43.63%
VOC2007 Trainval [2]	1200	82.16%	90.10%	45.11%
VOC2007 Test [2]	1200	82.47%	90.06%	45.22%
Arithmetic mean	1085.71	88.93%	86.97%	36.46%

Figure 4.5.4.5 shows the ROC curves generated for each database in this experiment. The performance of the STIMcoke database exceeded that of the other databases despite having far less of the image occupied by regions of interest (only 1.70% against an average of 36.46%). We attribute its superior results to the salient-by-design nature of the regions of interest in the image, a characteristic that is not the central focus of the other databases.

The high False Alarm Rate for all databases can be attributed to the Inhibition of Return (IOR), a component of the model of bottom-up visual attention we used. When a region is attended to its attention values are immediately suppressed (reduced) and slowly recover. This can be illustrated by the points in Figures 2 (c) and 3 (c). For example, in Figure 4.5.4.3 (c) the first point hits on a region of interest. This is suppressed and not hit again until the ninth point of attention (at which point the rest of the image has been suppressed and that region has recovered). This leads to an interesting situation where it is unlikely (indeed, nearly impossible if the region is small) to have two hits on the same region in a row, artificially increasing the False Alarm Rate considerably.

The results show that while using points of attention is suitable for most of the databases (STIMcoke, the VOC2005 databases, and the VOC2006 databases), the performance falls when applied to the largest and most diverse databases (both VOC2007 databases). This indicates that the nature of the dataset must be carefully considered before applying such a model.

4.5.4.6 Concluding Remarks

In this paper we tested a computational model of bottom-up human visual attention for detecting salient regions of interest in image databases. Seven databases were tested. We conclude that computational models of visual attention hold the potential for an unsupervised, biologically-inspired method of determining seeds for seed-based region-growing algorithms.

We set out to evaluate until what point in time it is reasonable to consider the salient points of attention (each of which is predicted for a certain time stamp) in addition to comparing the suitability of a variety of databases. The Itti-Koch model, like the human eyes, generates points serially. While it can run indefinitely, predictions after a certain time are not valuable as the model is only designed to approximate bottom-up visual attention -- the early processes of vision. The experiments presented in this paper confirm that the first few points of attention provide the most valuable information. The contribution of points of attention predicted at later points is biased by the inhibition of return.

Experiments reported in this paper demonstrate the validity of the proposed methodology, particularly when the objects of interest are salient by design. For example, the STIMcoke database, despite consisting of few and small regions yields the best results. The experiments demonstrate that performance is generally not affected by the number and size of regions of interest in the ground truth.

Ongoing work is focused on growing regions around the seeds generated by the model of visual attention used in these experiments and then comparing a variety of

segmentation algorithms to the results. The next stage is to extract features from the generated regions. The work will be incorporated as part of a complete CBIR system.

References for Section 4.5.4

[1] J. Davis and M. Goadrich. The relationship between precision-recall and roc curves. In ICML '06: Proceedings of the 23rd international conference on Machine learning, pages 233.240, New York, NY, USA, 2006.
[2] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascalnetwork.org/challenges/VOC/voc2007/workshop/index.html
[3] M. Everingham, A. Zisserman, C. K. I. Williams, and L. V. Gool. The 2006 pascal visual object classes challenge (voc2006) results. Technical report, University of Oxford, 2007.
[4] M. Everingham, A. Zisserman, C. K. I. Williams, L. van Gool, M. Allan, C. M. Bishop, O. Chapelle, N. Dalal, T. Deselaers, G. Dorko, S. Duffner, J. Eichhorn, J. D. R. Farquhar, M. Fritz, C. Garcia, T. Grif_ths, F. Jurie, D. Keysers, M. Koskela, J. Laaksonen, D. Larlus, B. Leibe, H. Meng, H. Ney, B. Schiele, C. Schmid, E. Seemann, J. Shawe-Taylor, A. Storkey, S. Szedmak, B. Triggs, I. Ulusoy, V. Viitaniemi, and J. Zhang. The 2005 pascal visual object classes challenge. In Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment (PASCAL Workshop 05), number 3944 in Lecture Notes in Arti_cial Intelligence, pages 117.176, Southampton, UK, 2006.
[5] L. Itti and C. Koch. Feature combination strategies for saliency-based visual attention systems. Journal of Electronic Imaging, 10(1):161.169, Jan 2001.
[6] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. IEEE Trans. on PAMI, 20(11):1254.1259, Nov 1998.
[7] D. Walther and C. Koch. Modeling attention to salient protoobjects. Neural Netw, 19(9):1395.407, 2006.

4.5.5 Survey of Recent Databases Suitable for Region-Oriented Content-Based Image Retrieval Applications

In this section we present a brief survey of several recently published image databases. Certain content-based image retrieval tasks are concerned with semantic regions of interest or individual objects within images rather than the images' global characteristics. It is essential to have a proper database with adequate ground truth information in order to evaluate these tasks. We discuss the evaluation criteria used when considering databases. Information on twenty databases is presented. Finally, we discuss the relative merits of several of the databases for specific applications.

4.5.5.1 Introduction

Content-based image retrieval (CBIR) is a fascinating field that has produced a wide variety of real-world applications. While some CBIR systems retrieve images based solely on global characteristics, other contemporary implementations try to interpret the content of images by extracting information from specific objects or regions of interest (ROIs) within the test images.

There are several factors which motivate the creation of standard, readily available databases for image retrieval. Most significantly, it is important to be able to benchmark and compare new systems and algorithms with existing work. A common database makes it more straightforward to compare results. Another contributing factor is the reuse of ground truth annotation information, which is generally time consuming to manually generate. Ground truth annotation is necessary for applications which try to retrieve objects within images.

Traditional CBIR databases may not contain appropriate ground truth information to be used in such experiments. For example, the UCID databases [15-16] provide ground truth information for image retrieval (images that should be retrieved in a query for a particular image), but do not provide image or object classes. Other efforts, such as Benchathlon [9] are not oriented towards retrieving images based on the objects contained within, although it was intended to make CBIR systems comparable with each other.

This paper briefly summarizes the characteristics of twenty databases that are suitable for CBIR applications which focus on the regions with semantic meaning within images (e.g. cars, dogs, etc.).

4.5.5.2 Criteria

There are a variety of considerations that must be taken into account when considering an image database. They are summarized in the following subsections.

Scope

The selection of a dataset, particularly for content-based image retrieval, is critical. The dataset (an individual image database or an aggregation of several image databases) defines the scope of the retrieval algorithms to be developed. While this work considers more general datasets (it is not considering a special-purpose CBIR task, such as detecting human faces in video surveillance footage), dataset selection is still important. A database that is too narrow in scope may prevent the created methods from being extended to more general tasks, while a database that is too broad may make the retrieval problem intractable.

Database Size

There is a wide diversity of database sizes used in CBIR literature. Datasets as small as dozens of images have been used in CBIR tests, while ones as large as half a million images have recently been experimented with [20]. The amount of images on the Internet is staggering. For example, the Picsearch image search engine claims to have indexed over 1,700,000,000 (over 1.7 *billion*) pictures from the Internet (as of November 5, 2007) [12]. As stated previously, the image retrieval task must be well-bounded enough to be tractable, making extremely large databases ill-suited for experimentation. Table 4.5.5.1 shows that the truncated mean of the 20 image databases considered is approximately 3403 images, which is acceptable for many initial experiments in the CBIR community, before scaling is a consideration.

Number of Labeled Images

Ideally, every image in the database will have been manually annotated, enabling a wide variety of experiments and evaluation to be performed on the dataset.

Table 4.5.5.1 shows the number of images which have some sort of annotation associated. This annotation may be a general class for the entire image, labels given to one or more objects/regions of interest in the image, or a list or related images (correct responses in a content-based query).

Annotation Type

Databases containing images with images classified only on a global level of granularity (e.g. a database divided into image categories) and without specific object-based annotation are given the label ``None'' in the *Annotation* column in 1. Datasets which provide a bounding box associated with an object's label (e.g. two sets of coordinates) are labeled as ``Box'' in the same column. An example of an image with bounding box annotation is shown in Figure 4.5.5.1. More detailed than a bounding box is a database labeled with more than two coordinates per image, resulting in a bounding ``Polygon'' rather than a bounding box, as shown in Figure 4.5.5.2. The most precise annotation is a ``Mask'' of each image in the database. An example of an image and a mask for an object within the image is shown in Figure 4.5.5.3. Polygon or mask annotation can be reduced to bounding box annotation.

Number of Annotated Objects

If exactly one object has been annotated in each image in the database, this number will be the same as the number of images in the database. It can be higher than the number of images in the database if more than one object, on average, has been annotated in each image. It may be lower than the number of images in the database if annotation is incomplete.

The truncated mean of the number of annotated objectes in the considered image databases is approximately 6306, indicating that most images have at least one object in them (the actual percentage of images with objects in them is lower as many images contain more than one object). Our objective is to use databases where the number of annotated objects is greater than or equal to the number of images in the database.

Number of Classes

This is the number of different categories for images (if only global labeling is given) or the number of different objects in the database (if available). This is different than *Objects*, in that an image may contain a lion, a tiger, and a bear (three objects), but still fall under a single class (e.g. ``animals'').

Image or object classes may vary from extremely general (e.g. indoor, outdoor), to broad (e.g. people, sports, vehicles), to more narrow categories (e.g. boats, planes, cars). A greater number of image/object classes leaves less room to differentiate between the individual classes, and thus makes the content-based retrieval task more

difficult. It is also preferable that the classes be fairly evenly balanced, that is, they categories each refer to a similar number of images or objects.

Density

Density, $D(i,o)$, is the average density of annotated objects within each image. In other words, it is the total number of annotated objects in the database divided by the total number of annotated images in the database:

$$D(i,o) = \frac{\sum_{i}^{n} An_{io}}{\sum_{i}^{n} A(i)} \tag{1}$$

$$A(i,o) = \begin{cases} 1, & An_{io} > 0 \\ 0, & Else \end{cases} \tag{2}$$

where An_{io} is the number of annotated objects o in image i and n is the number of images in the database. $A(i,o)$ is a function that helps count the total number of images which have annotation. It returns 1 if An_{io} is greater than or equal to 1 (i.e. the image contains at least one annotated object) and 0 otherwise (i.e. the image does not contain any annotated objects).

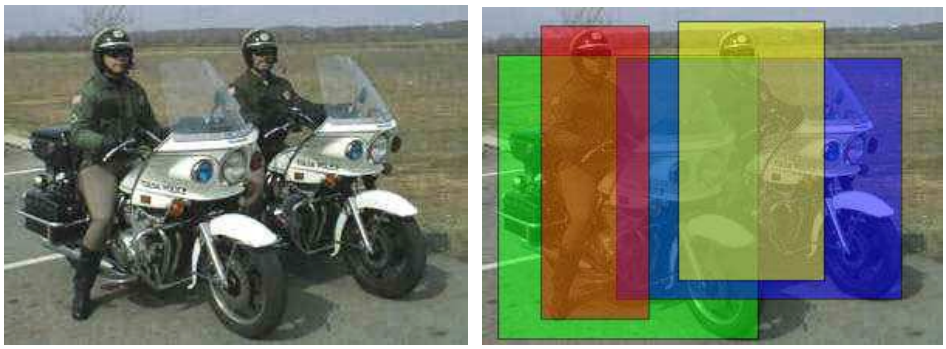


Figure 4.5.5.1. Ground truth bounding boxes.

Figure 4.5.5.2. Ground truth polygons.

Figure 4.5.5.3. An image and its object masks.

Table 4.5.5.1. Comparison of image databases.

Name	Images	Labeled	Annotation	Objects	Classes	Density
Caltech [5]	5775	4620	Box	1293	6	0.28
Caltech 101 [4]	9197	9197	None		101	
FAU Salient [8]	1471	1471	None		12	
LabelMe [14]	160569	41221	Polygon	Many	Many	
MIT-CSAIL [18]	72000	2873	Polygon	10358	Many	3.61
MSRC OCV1 [17]	240	240	Mask		9	
MSRC OCV2 [17]	591	591	Mask		23	
MSRC ORID [17]	4322	4322	None		33	
Renninger [13]	995	995	None		10	
Simplicity1000 [19]	1000	1000	None		10	
Simplicity10000 [19]	10000	10000	None		100	
STIMautobahn [6]	180	180	Mask			
STIMcoke [6]	208	208	Mask			
TU Darmstadt [7]	327	326	Box	336	3	1.03
TU Graz-02 [11]	1476	1280	Mask	1816	4	1.42
VOC2005 1 [3]	1578	1578	Box	2209	4	1.40
VOC2005 2 [3]	654	654	Box	1293	4	1.98
VOC2006 Trainval [2]	2618	2618	Box	5455	10	2.08
VOC2006 Test [2]	2686	2686	Box	4052	10	1.51

VOC2007 Trainval [1]	9963	9963	Box	24640	20	2.47
VOC2007 Test [1]	4953	4953	Box	15662	20	3.16
Arithmetic mean	13847.76	4080.38		6711.40	22.29	1.89
Truncated mean	3402.71	3323.18		6306.22	22.29	1.70

4.5.5.3 Databases

Table 4.5.5.1 shows a summary of the twenty databases considered. The arithmetic mean is the sum all defined valued divided by the number of values in the former summation. The truncated mean is the same as the previously-defined arithmetic mean except the two largest and two smallest values are removed from consideration (and thus the number of values considered is reduced by four). This was done to stop extremely large (or small) outlying values from improperly skewing the results -- those from the LabelMe [14], MIT-CSAIL [18], STIMautobahn [6], and STIMcoke [6] databases.

A variety of databases were considered for this work. This evaluation is summarized in Table 4.5.5.1. Some brief notes on each of the databases follow:

- **Caltech [5]**: consists of mostly vehicles such as cars, planes, and motorcycles. Some of the images do not contain objects of interest.
- **Caltech 101 [4]**: images are divided into 101 semantic categories, although individual objects are not considered. The categories are not evenly distributed (the largest refers to 800 images, whereas the smallest has only 31 images).
- **FAU Salient [8]**: a database of 1471 images consisting mostly of signs, sports balls, and other salient objects. Pictures were taken both indoors and outdoors at different times of day. Most of the photos were designed so that the image contains a single regions of interest, although a variety of distractors are present throughtout. Additionally, there is a subset of images with multiple target regions of interest.
- **LabelMe [14]**: a massive database of manually-labeled objects. The LabelMe database is an order or magnitude larger than any other database considered. Interactive tools allow human users to manually draw polygons around regions or objects in the image (e.g. both a road and a car may be annotated). The database accepts submitted annotation, and, as a result, continues to become more complete.
- **MIT-CSAIL [18]**: a large image database, but only a small fraction of the images are labeled. The object class sizes ranges from as small as 1 to as large as 693. In addition to the 107 object classes provided in the ground truth, 18 region classes, such as ``floor'' or ``sky'' are also distinguised.
- **MSRC OCV1 [17]**: a small, fully-annotated database of images with ground truth provided as manually-generated pixel-precise masks. Categories include bicycles, cars, cows, airplanes, people, and several outdoor scenes without objects of interest.

- **MSRC OCV2 [17]**: similar to MSRC OCV1, but over twice as large.
- **MSRC ORID [17]**: thousands of images are grouped into semantic categories, although no object-specific annotation is provided.
- **Renninger [13]**: images are grouped into semantic categories. The images are general scenes without consideration of the specific objects within them, making the database more suitable for CBIR based on global characteristics.
- **Simplicity1000 [19]**: images are grouped into semantic categories, although no object-specific annotation is provided.
- **Simplicity10000 [19]**: this database shares the same characteristics as Simplicity1000, except it is ten times as large and has ten times as many classes.
- **STIMautobahn [6]**: a small set of images with salient objects (road signs and markers). Ground truth is provided as manually-generated pixel-wise masks.
- **STIMcoke [6]**: a small set of images with a salient soda can in each one. Ground truth is as in STIMautobahn.
- **TU Darmstadt [7]**: the database provides three object categories: cars, cows, and motorcycles. The database is completely annotated except for one image.
- **TU Graz-02 [11]**: four object categories are provided, although one consists of images with no objects or regions of interest.
- **VOC2005 1 [3]**: the PASCAL Object Recognition Database Collection [10] is an effort to standardize the annotation (ground truth) of image databases. An annual competition is held to compare various image retrieval systems. Many of the considered databases are part of the collection. Because the annotation is uniform across the databases, databases that are part of this collection may be preferred. The databases in this survey that are part of the PASCAL collection are Caltech, Caltech 101, MIT-CSAIL, TU Darmstadt, TU Graz-02, and the VOC Challenge databases. In this particular database the images have been divided into four classes: bicycles, cars, motorcycles, and people. Images in the database have been compiled from other image databases cited in this list.
- **VOC2005 2 [3]**: a second database for the VOC2005 challenge. It has the same characteristics as the first VOC2005 database.
- **VOC2006 Trainval [2]**: the images in this database are from flickr [21] and Microsoft Research Cambridge [17]. Objects may fall into one of ten object classes: bicycles, buses, cats, cars, cows, dogs, horses, motorbikes, people, and sheep.
- **VOC2006 Test [2]**: a test database for the VOC2006 challenge. It has the same characteristics as the first VOC2006 database.
- **VOC2007 Trainval [1]**: the database for the 2007 VOC competition is similar to the one from the previous years, except that the number of images has increased, as has the number of classes making the database more challenging. Annotation data is now in XML rather than text files.
- **VOC2007 Test [1]**: a test database for the VOC2007 challenge. It has the same characteristics as the first VOC2007 database.

4.5.5.4 Concluding Remarks

In this section we defined several requirements and preferences for the image databases in the context of region-based CBIR. In summary, the most important requirements for dataset selection are:

- Individually-annotated objects -- object in the images should be individually labeled
- Each image should have at least one labeled object (density greater than or equal to 1.00)
- The database should have a reasonable number of fairly-balanced object classes
- The database should be a reasonable size for the given image retrieval task

Traditional CBIR applications need to consider the very largest databases available, on the order of 10000 images or higher, leaving only LabelMe [14], MIT-CSAIL [18], and perhaps Simplicity10000 [19]. However, only LabelMe provides annotation for significantly more than 10000 images.

For initial testing, the small databases may be more appropriate, as the quality of their ground truth (masks of the relevant objects) is of much higher quality than simply categories or even coarse bounding boxes.

In our work to retrieve images based on their salient object we have found that a good middle ground are the databases from the VOC Challenge [reference redacted]. They are completely annotated with bounding boxes manually created for each image, and are restricted to a reasonable number of object classes.

Researchers have more options than ever before in regards to using freely available (and thus, readily comparable) databases in their projects. Using the wrong dataset can be perilous to nascent research and has the potential to derail results early on. Additionally, the database must be challenging enough to properly validate the research.

We are satisfied with the variety of databases currently available. Furthermore, projects which are ongoing (such as LabelMe [14]) improve with time, constantly adding new annotation from users. In this case, it would be desirable to have a fully annotated, non-changing subset of the LabelMe dataset to benchmark CBIR applications.

Researchers should not have to create their own databases (and thus preclude comparison to their work) if they are creating a general-purpose object-based CBIR system. However, a greater variety of larger, fully-annotated databases would be welcome.

References for Section 4.5.5

[1] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascalnetwork.org/challenges/VOC/voc2007/workshop/index.html.
[2] M. Everingham, A. Zisserman, C. K. I.Williams, and L. V. Gool. The 2006 pascal visual object classes challenge (voc2006) results. Technical report, University

of Oxford, 2007.
[3] M. e. a. Everingham. The 2005 pascal visual object classes challenge. In Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment (PASCAL Workshop 05), number 3944 in Lecture Notes in Artificial Intelligence, pages 117-176, Southampton, UK, 2006.
[4] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples an incremental bayesian approach tested on 101 object categories. In Proceedings of the Workshop on Generative-Model Based Vision, Washington, DC, June 2004.
[5] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, volume 2, pages 264-271, Madison, Wisconsin, June 2003.
[6] L. Itti and C. Koch. Feature combination strategies for saliency-based visual attention systems. Journal of Electronic Imaging, 10(1):161-169, Jan 2001.
[7] B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit shape model. In Proceedings of the Workshop on Statistical Learning in Computer Vision, Prague, Czech Republic, May 2004.
[8] O. Marques, L. M. Mayron, G. B. Borba, and H. R. Gamba. An attention-driven model for grouping similar images with image retrieval applications. EURASIP Journal on Advances in Signal Processing, 2007:Article ID 43450, 17 pages, 2007. doi:10.1155/2007/43450.
[9] H. M¨uller, W. M¨uller, D. M. Squire, S. Marchand-Maillet, and T. Pun. Performance evaluation in content-based image retrieval: overview and proposals. Pattern Recogn. Lett., 22(5):593-601, 2001.
[10] P. Network. The pascal object recognition database collection.
[11] A. Opelt, M. Fussenegger, A. Pinz, and P. Auer. Generic object recognition with boosting. Technical Report TR-EMT-2004-01, EMT, TU Graz, Austria, 2004. Submitted to the IEEE Transactions on Pattern Analysis and Machine Intelligence.
[12] Picsearch. Picsearch - image search for pictures and images.
[13] L. W. Renninger and J. Malik. When is scene identification just texture recognition? Vision Research, 44(19):2301-2311, September 2004.
[14] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: a database and web-based tool for image annotation. MIT AI Lab Memo AIM-2005-025, 2005.
[15] G. Schaefer and M. Stich. UCID - An Uncompressed Colour Image Database. Technical report, School of Computing and Technology, The Nottingham Trent University, Nottingham, United Kingdom, 2003.
[16] G. Schaefer and M. Stich. Ucid - an uncompressed colour image database. In Storage and Retrieval Methods and Applications for Multimedia 2004, volume 5307 of Proceedings of SPIE, pages 472-480, 2004.
[17] J. Shotton, J.Winn, C. Rother, and A. Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In Proceedings of European Conference Computer Vision (ECCV), 2006.
[18] A. Torralba, K. P. Murphy, and W. T. Freeman. Sharing features: efficient boosting procedures for multiclass object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, volume 2, pages 762-769, Washington, DC, June 2004.
[19] J. Z. Wang, J. Li, and G. Wiederhold. Simplicity: Semantics-sensitive integrated matching for picture libraries. In VISUAL '00: Proceedings of the 4th International

Conference on Advances in Visual Information Systems, pages 360-371, London, UK, 2000. Springer-Verlag.
[20] Y. M. Wong, S. C. H. Hoi, and M. R. Lyu. An empirical study on large-scale content-based image retrieval. In Multimedia and Expo, 2007 IEEE International Conference on, pages 2206-2209, July 2007.
[21] Yahoo! Welcome to flickr - photo sharing.

4.5.6 Investigation of the Suitability of Using a Computational Model of Visual Attention for Detecting Objects of Interest in Video Surveillance Footage

In section work we investigate the suitability of using points of attention generated by a computational model for detecting objects of interest in video surveillance footage. The computational model was used to generate points of attention for thirty video surveillance sequences. The results were then analyzed and discussed, with specific recommendations made as to how to improve performance. We recommend empirically determining a threshold of the voltage (intensity) of points to discard and creating a post-processing block for discarding points that target areas of the frame that do not exhibit movement.

4.5.6.1 Introduction and Background

To evaluate the suitability of using the seeds generated by a computational model of visual attention to detect objects of interest in video surveillance sequences we used the Itti-Koch model of visual attention [2]. The Itti model has been refined and used in a wide variety of applications. There are implementations of the model publicly available in several programming languages. It was employed to generate ten points of attention for each image in the dataset. The objective was to record a generous number of points of attention which would be predicted for times that are far beyond reasonable application of the model (times that exceed those of bottom-up attention). As the model is bottom-up, it was designed only to represent the first few saccades of the human vision system, before top-down factors begin to influence the vision task [3].

We were interested in observing the behavior of the model with regards to video surveillance footage. The model takes a sequence of video frames as input. The output we recorded for each shift in attention includes the x and y coordinates of the point, the predicted time, the frame number (derived from the predicted time), and the intensity of the attentional peak shifted to (in millivolts - mV). This work empirically determines if there is a threshold beyond or before the intensity of the voltage of the point of attention reduces performance. Determining a reasonable point to cutoff results will help make the system efficient. We used receiver operating characteristic (ROC) curves [1] for this assessment.

4.5.6.2 Dataset

The experiments were performed using a subset of thirty sequences from CAVIAR video sequences [1]. The selected video sequences are:

- br1gt: a person pacing
- br2gt: a person pacing and pausing

- br3gt: a person pacing and reading
- br4gt: a person by a reception desk
- bww1gt: a person waiting
- bww2gt: a person waiting
- fcgt: two people fight and run after each other
- fomdgt1: two people fight, one falls, one flees
- fomdgt2: two people fight, one falls, one flees
- fomdgt3: two people fight, one falls, one flees
- fra1gt: two people fight and separate
- fra2gt: two people fight and separate
- lb1gt: a person leaving a bag behind
- lb2gt: a person leaving a bag behind
- lbbcgt: a person leaving a bag behind
- lbgt: a person leaving a box behind
- lbpugt: a person leaving a bag and later retrieving it
- mc1gt: four people meet and separate
- ms3ggt: two people encounter a third and separate
- mws1gt: two people encounter each other and separate
- mwt1gt: two people encounter each other and walk
- mwt2gt: two people encounter each other and walk
- rffgt: a person falling down
- ricgt: a person resting in a chair
- rsfgt: a person on the floor
- rwgt: a person moving on the floor
- spgt: two people enter a scene together and separate
- wk1gt: one person walking in a straight line
- wk2gt: one person walking in a straight line and back
- wk3gt: one person walking

All sequences are color and represent surveillance footage. A sample frame is shown in Figure 4.5.6.1. The security camera is mounted in the corner on the ceiling with a wide-angle lens. Three people (object of interest) are shown (bounded by yellow boxes). The green box encircles a ground of people but is not used in this work.

Figure 4.5.6.1. Sample frame including ground truth bounding box information. (image from [1])

4.5.6.3 Experiments

The iLab Neuromorphic Vision C++ Toolkit (iNVT, also known as Ezvision) [2] was used to calculate points of attention in our experiments. While there are Java and MATLAB implementations of the computational model of visual attention that have more accessible and compact code, the C++ implementation was a necessity for this work. It is the authoritative implementation and the most up-to-date. It is, by far, the fastest to execute, which is a necessity when dealing with video sequences with thousands of frames. It is also the only implementation that can handle video sequences natively (as a sequence of consecutively-numbered images).

While the iNVT was designed for Linux, we were able to successfully compile and run the toolkit on several Windows XP workstations using the Cygwin environment as an intermediary layer. Many intermediate packages are required before the iNVT will fully compile.

The ground truth consists of labeled bounding boxes of target objects. Each object is uniquely labeled, so a person talking, even if they are occluded or stop walking at a certain point, will always be labeled as the same object.

Points of attention were extracted for each video sequence. Then, the extracted points were compared with the ground truth to see if they fell within the bounding box of a target object. The possible outcomes are as follows:

- True positive: this is the case where a point of attention falls within the bounding box of an object. One a unique object is hit it cannot be counted as a true positive again. Subsequent points of attention which hit the object are not included in any counts.
- False positive: a point of attention that falls outside of a target object in the same frame is a false positive – an undesirable result.
- False negative: a false negative is an object that is never hit by a point of attention – one that eludes detection

The fourth possible case, false positives, does not apply to this work, as there are no areas of the sequences that can be identified as such. The total number of points of attention was recorded as well.

For each ground truth object the following information was noted:

- The frame the object first appears in the ground truth
- The time the object first appears in the ground truth
- The frame the object last appears in the ground truth
- The time the object last appears in the ground truth
- The frame the object is first hit by a point of attention
- The time the object is first hit by a point of attention
- The difference, in frames, between the object first appearing in the ground truth and the object first being hit by a point of attention
- The difference, in time, between the object first appearing in the ground truth and the object first being hit by a point of attention
- The total number of frames the object is present in the ground truth
- The total time the object is present in the ground truth
- The number of times the object is hit be a point of attention

4.5.6.4 Results and Discussion

Results are illustrated as a receiver operating characteristic (ROC) curve. An ROC curve plots the Hit Rate vs. the False Alarm Rate. In an ideal case, the ROC curve will be close to the top left corner of the plot, where the Hit Rate is high and the false positive rate is low. An inflection point on the ROC curve indicates a point where returns begin to diminish (i.e. beyond where the samples do not contribute to the results). The voltage associated with the sample at that inflection point is recorded.

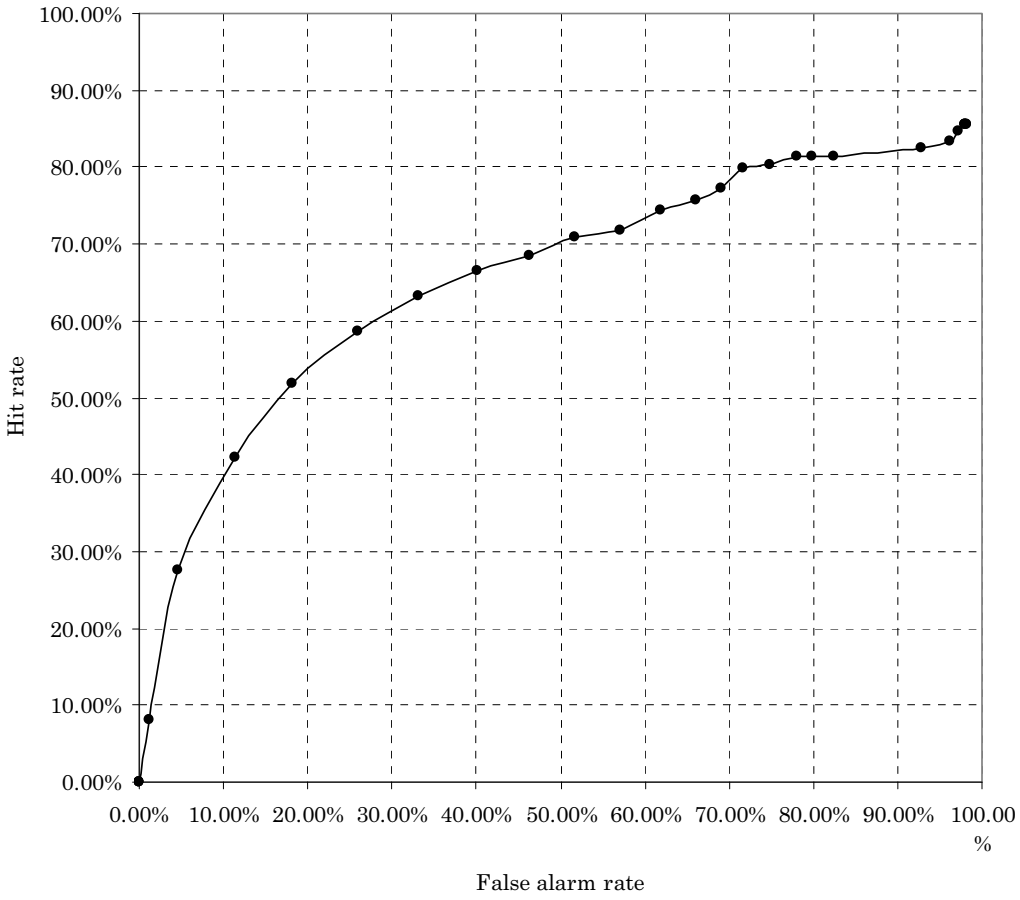


Figure 4.5.6.2. Receiver operating characteristics for 30 sequences.

Figure 4.5.6.2 shows the ROC curve generated across the 30 sequences by varying the voltage cutoff. In this graph the maximum false alarm rate has been kept constant. The majority of the graph is above the point where Hit rate = False alarm rate. The hit rate never reaches 100% as there are objects which are never hit by the points of attention.

The Hit rate is defined as follows:

$$HitRate_{roi} = \frac{TP_{roi}}{TP_{roi} + FN_{roi}}$$

The False alarm rate is defined as follows:

$$FalseAlarmRate_{poa} = \frac{FP_{poa}}{MaxFP_{poa}}$$

Table 4.5.6.1 shows the data which was used to construct this curve:

Table 4.5.6.1. Hit rate and false alarm rate.

mV	Hit Rate	False Alarm Rate	MaxFA
0	0.00%	0.00%	0.00%
0.1	0.00%	0.00%	0.00%
0.2	0.00%	0.00%	0.00%
0.3	8.07%	0.00%	1.12%
0.4	27.54%	88.54%	4.61%
0.5	42.20%	92.41%	11.36%
0.6	51.75%	94.24%	18.07%
0.7	58.71%	95.33%	25.86%
0.8	63.32%	96.05%	33.16%
0.9	66.45%	96.53%	40.18%
1	68.49%	96.82%	46.34%
1.1	**70.99%**	**97.08%**	**51.74%**
1.2	71.69%	97.31%	57.00%
1.3	74.50%	97.40%	61.82%
1.4	75.76%	97.52%	65.99%
1.5	77.29%	97.57%	68.97%
1.6	79.79%	97.59%	71.76%
1.7	80.34%	97.68%	74.92%
1.8	81.46%	97.76%	77.94%
1.9	81.46%	97.81%	79.86%
2	81.46%	97.88%	82.42%
3	82.54%	98.09%	92.81%
4	83.37%	98.13%	96.14%
5	84.71%	98.12%	97.30%
6	85.54%	98.11%	97.92%
7	85.54%	98.11%	98.00%
8	85.54%	98.12%	98.10%
9	85.54%	98.12%	98.12%
10	85.54%	98.12%	98.12%

It maps the average hit rate and false alarm rate against voltage. False alarm rate is calculated both in terms of the number of attempts before and after applying the voltage filter. The highlighted point, 1.1mV is the empirically-determined inflection point – the cutoff above which gains in hit rate are outpaced by larger increases in the false alarm rate.

We performed an in-depth analysis of the “wk1gt” sequence. It is composed of 610 frames. It contains four ground truth objects. A total of 297 points of attention were generated for this sequence, with 44 of them hitting the objects of interest in the sequence. Eleven metrics were either recorded or calculated. Table 4.5.6.2 shows a summary of the recorded data:

Table 4.5.6.2. Summary of results for the "wk1gt" sequence.

Object ID	0	1	4	5	Average
gtfirst_f	17	39	63	236	**88.75**
gtfirst_t	680	1560	2520	9440	**3550**
gtlast_f	179	610	610	511	**477.5**

gtlast_t	7160	24400	24400	20440	**19100**
mfirst_f	23	90	114	250	**119.25**
mfirst_t	949.4	3630.6	4568.4	10018.7	**4791.775**
Diff_first_f	6	51	51	14	**30.5**
Diff_first_t	269.4	2070.6	2048.4	578.7	**1241.775**
Diff_gt_f	162	571	547	275	**388.75**
Diff_gt_t	6480	22840	21880	11000	**15550**
Count	21	5	1	17	**11**

The following is the legend for the labels in Table 4.5.6.2:

- gtfirst_f: the first frame the object appears in the ground truth
- gtfirst_t: the time at which the object first appears in the ground truth
- gtlast_f: the last frame the object appears in the ground truth
- gtlast_t: the last time the object appears in the ground truth
- mfirst_f: the frame in which the first point of attention lands on the object
- mfirst_t: the time at which the first point of attention lands on the object
- diff_first_frame: the delay, in frames, between the object first appearing in the ground truth and being hit by a point of attention
- diff_first_time: the delay, in time, between the object first appearing in the ground truth and being hit by a point of attention
- diff_gt_f: the duration, in frames, of the object in the ground truth
- diff_gt_time: the duration, in time, of the object in the ground truth
- count: the total number of time the object is hit by a point of attention

A number of interesting observations can be made from this data. The average object exists for 19100ms in the sequence. During that time it is hit, on average, 11 times (about 1736ms between hits). A delay of, on average, 1272ms exists between the object presenting itself and it being detected by a point of attention.

4.5.6.5 Concluding Remarks

The experiments performed show that a computational model of visual attention is able to target objects of interest in surveillance video sequences. One challenge is the high amount of false positives. Our filter, adjusting the threshold of the voltage (intensity) of points that are recorded is able to dramatically reduce the amount of false positives, but there is still room for improvement. We recommend an additional post-processing block that is able to exclude points that occur where no movement is occurring.

References for Section 4.5.6

[1] http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1/
[2] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. IEEE Trans. on PAMI, 20(11):1254–1259, Nov 1998. 1, 2, 3
[3] J. Davis and M. Goadrich. The relationship between precision-recall and roc curves. In ICML '06: Proceedings of the 23rd international conference on Machine learning, pages 233–240, New York, NY, USA, 2006. ACM. 3

4.6 Summary of Contributions and Deliverables

The deliverables for CCST year 3 research are as follows:

- Technical report that describes in detail our research methodology, results and contributions.
- Software programs (with source code) that implement the proposed algorithms.